DEVELOPMENT OF A LIQUID JET MODEL FOR IMPLEMENTATION IN A 3-DIMENSIONAL EUQLARIAN ANALYSIS TOOL

A Dissertation in
Nuclear Engineering

by
Francis X Buschman III

© 2008 Francis X Buschman III

Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

August 2008
The dissertation of Francis X Buschman III was reviewed and approved* by the following:

Lawrence Hochreiter  
Professor of Mechanical and Nuclear Engineering  
Dissertation Co-Advisor  
Co-Chair of Committee

John Mahaffy  
Associate Professor of Nuclear Engineering  
Dissertation Co-Advisor  
Co-Chair of Committee

Fan-Bill Cheung  
Professor of Mechanical and Nuclear Engineering

Cengiz Camci  
Professor of Aerospace Engineering

David Aumiller  
Fellow Engineer, Bechtel Bettis  
Special Member

Jack Brenizer  
Professor of Mechanical and Nuclear Engineering  
Chair of Nuclear Engineering Program

*Signatures are on file in the Graduate School
ABSTRACT

The ability to model the thermal behavior of a nuclear reactor is of utmost importance to the reactor designer. Condensation is an important phenomenon when modeling a reactor system’s response to a Loss Of Coolant Accident (LOCA). Condensation is even more important with the use of passive safety systems which rely on condensation heat transfer for long term cooling. The increasing use of condensation heat transfer, including condensation on jets of water, in safety systems puts added pressure to correctly model this phenomenon with thermal-hydraulic system and sub-channel analysis codes.

In this work, a stand alone module with which to simulate condensation on a liquid jet was developed and then implemented within a reactor vessel analysis code to improve that code’s handling of jet condensation. It is shown that the developed liquid jet model vastly improves the ability of COBRA-TF to model condensation on turbulent liquid jets.

The stand alone jet model and the coupled liquid jet COBRA-TF have been compared to experimental data. Jet condensation heat transfer experiments by Celata et al with a variety of jet diameters, velocities, and subcooling were utilized to evaluate the models. A sensitivity study on the effects of noncondensables on jet condensation was also carried out using the stand alone jet model.
# TABLE OF CONTENTS

**LIST OF FIGURES** ............................................................................................................ v

**LIST OF TABLES** .......................................................................................................... vii

**NOMENCLATURE** ......................................................................................................... viii

**ACKNOWLEDGEMENTS** ............................................................................................... xiv

**Chapter 1  Introduction** ................................................................................................ 1

  1.1 Problem Description ............................................................................................... 6
  1.2 Originality of Current Work ................................................................................. 11

**Chapter 2  Experiments on Condensation Heat Transfer on Liquid Jets** .............. 12

**Chapter 3  Development of the Stand-alone Jet Computational Model** .............. 18

  3.1 Background for Equations of Motion for Liquid Jet .......................................... 19
  3.2 Equations of Motion for Liquid Jet ........................................................................ 21
  3.3 Equations of Motion for Vapor Boundary ............................................................ 23
  3.4 Required Constitutive Relationships ..................................................................... 26
     3.4.1 Jet Breakup ....................................................................................................... 26
     3.4.2 Interfacial Heat Transfer ................................................................................... 27
     3.4.3 Sensible Heat Transfer ....................................................................................... 34
     3.4.4 Turbulence within the Liquid Jet ..................................................................... 35
     3.4.5 Jet Formation .................................................................................................... 40
     3.4.6 Jet Termination .................................................................................................. 40
     3.4.7 Effects of Noncondensable Gas on Condensation ......................................... 41
     3.4.8 Interfacial Shear ............................................................................................ 46
  3.5 Summary of Selected Constitutive Models ......................................................... 47

**Chapter 4  Linearization and Implementation of Equations of Motion** ............... 49

  4.1 Axial Momentum Equation Linearization ............................................................ 52
  4.2 Radial Momentum Equation Linearization .......................................................... 60
  4.3 Continuity Equation Linearization ........................................................................ 65
  4.4 Energy Equation Linearization ............................................................................. 75
  4.5 Non-Condensable Gas Equation Linearization ................................................... 88
  4.6 Boundary Conditions ............................................................................................ 95
  4.7 Verification of Stand Alone Liquid Jet Model ...................................................... 98

**Chapter 5  Implementation of Jet Model in COBRA-TF** ....................................... 105

  5.1 COBRA-TF Conservation Equations ..................................................................... 107
5.2 COBRA-TF Finite Difference Equations .......................................................... 109
5.3 Changes to Conservation Equation for Liquid Jet Model ............................... 114
5.4 Coupling of Jet Model to COBRA-TF .............................................................. 116
5.5 Verification of Jet Model Coupling to COBRA-TF ......................................... 118

Chapter 6 Discussion of Results ............................................................................. 123

6.1 Comparison of Liquid Jet Model to Test Data .............................................. 123
6.2 Turbulent Jet Results ..................................................................................... 125
6.3 Turbulence modeling ..................................................................................... 142
6.4 Laminar Jet Results ...................................................................................... 145
6.5 Transition Jet Results ................................................................................... 154
6.6 Validation of Liquid Jet Model Coupling with COBRA-TF ......................... 157

Chapter 7 Conclusions and Recommendations for Future Work ....................... 162

Bibliography ............................................................................................................. 167

Appendix A Experimental Data from Celata et al Test ........................................... 171

Appendix B Listing of Liquid Jet Model ................................................................. 201

B.1 Liquid_jet.f90 ............................................................................................... 201
B.2 Coupling Subroutine (liquid_jet_int.f90) ..................................................... 267
LIST OF FIGURES

Figure 1-1: Control Volumes [6] ..............................................................................................................3
Figure 1-2: Boundary Layer Surrounding Liquid Jet ...........................................................5
Figure 1-3: Physical Picture of Liquid Jet injected into Vapor Environment.....................7
Figure 1-4: Liquid/Vapor Interface with Gas Accumulation.................................................10
Figure 2-1: Test Section for Jet Condensation Experiment[19] ..............................................13
Figure 2-2: Uncertainty in Celata Jet Condensation Test ..................................................17
Figure 3-1: Physical Picture of Condensation Process on a Cold Wall [34] ......................42
Figure 4-1: Two-Dimensional depiction of a computational cell and its neighbors...50
Figure 4-2: Axial momentum staggered mesh.................................................................51
Figure 4-3: Radial momentum staggered mesh .....................................................................51
Figure 4-4: Noding for simulation of jet with no heat and mass transfer and no interfacial shear..............................................................99
Figure 4-5: Radial Velocity Profile in Liquid Jet with no Heat and Mass Transfer and no Interfacial Shear .............................................................................................................100
Figure 4-6: Noding of 3x3 problem .....................................................................................101
Figure 5-1: Sketch of jet model within COBRA-TF mesh.................................................106
Figure 5-2: Total vessel mass versus time ........................................................................119
Figure 5-3: Energy versus Time ............................................................................................121
Figure 6-1: Radial profile of the turbulent axial velocity at the 6 inch location...........126
Figure 6-2: Turbulent enthalpy radial profile at various axial locations .....................127
Figure 6-3: Comparison of Simulation of Run 1.002 to Experimental Data .................129
Figure 6-4: Comparison of Simulation of Run 1.003 to Experimental Data ...............131
Figure 6-5: Comparison of Simulation of Run 1.006 to Experimental Data ...............133
Figure 6-6: Comparison of Simulation of Run 1.200 with Experimental Data ...........135
Figure 6-7: Comparison of Simulation of Run 1.202 with Experimental Data ........... 136
Figure 6-8: Comparison of Simulation of Run 1.013 to Experimental Data .............. 138
Figure 6-9: Comparison of Simulation of Run 1.016 to Experimental Data ............. 139
Figure 6-10: Comparison of Simulation of Run 1.211 to Experimental Data .......... 140
Figure 6-11: Comparison of Simulation of Run 1.215 to Experimental Data .......... 141
Figure 6-12: Isachenko and Travis Models for Eddy Diffusivity Ratio ................. 144
Figure 6-13: Radial profile of the laminar jet axial velocity at the 6 inch location .... 146
Figure 6-14: Laminar jet enthalpy radial profile at several axial locations .......... 147
Figure 6-15: Comparison of Simulation of Run 1.220 with Experimental Data .... 149
Figure 6-16: Comparison of Simulation of Run 1.221 with Experimental Data .... 150
Figure 6-17: Comparison of Simulation of Run 1.021 with Experimental Data .... 152
Figure 6-18: Comparison of Simulation of Run 1.011 with Experimental Data .... 153
Figure 6-19: Comparison of Simulation of Run 1.121 with Experimental Data .... 155
Figure 6-20: Comparison of Simulation of Run 1.110 with Experimental Data .... 156
Figure 6-21: Comparison of Simulation of Run 1.002 using both Stand Alone and Coupled Liquid Jet Models ................................................................. 159
Figure 6-22: Comparison of Simulation of Run 1.002 using Stand Alone and Coupled Liquid Jet Models and COBRA-TF ............................................................ 161
LIST OF TABLES

Table 2-1: Range of parameters for Celata test ........................................................... 16
Table 3-1: Correlations for Stanton Number from Previous Works......................... 29
Table 3-2: Correlations for Stanton number with Multiple Regions ....................... 30
Table 3-3: Summary of constitutive models................................................................. 48
Table 4-1: Mass Balance of Vapor Boundary with No Heat and Mass Transfer ....... 102
Table 4-2: Mass and Energy Balance for Vapor Boundary Layer........................... 103
Table 4-3: Mass and Energy Balance in the Liquid Jet ............................................. 104
# Nomenclature

## Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>LWR</td>
<td>Light Water Reactor</td>
</tr>
<tr>
<td>LOCA</td>
<td>Loss of Coolant Accident</td>
</tr>
<tr>
<td>PWR</td>
<td>Pressurized Water Reactor</td>
</tr>
</tbody>
</table>

## Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>Constant in equation for $k_g$</td>
</tr>
<tr>
<td>$A_{HT}$</td>
<td>Heat transfer area</td>
</tr>
<tr>
<td>$B$</td>
<td>Constant in equation for $k_g$</td>
</tr>
<tr>
<td>$C$</td>
<td>Constant in equation for $k_g$</td>
</tr>
<tr>
<td>$c^0$</td>
<td>Coefficient in axial momentum equation</td>
</tr>
<tr>
<td>$c^1$</td>
<td>Coefficient in axial momentum equation</td>
</tr>
<tr>
<td>$c^2$</td>
<td>Coefficient in axial momentum equation</td>
</tr>
<tr>
<td>$c_p$</td>
<td>Specific heat at constant pressure</td>
</tr>
<tr>
<td>$d$</td>
<td>Nozzle diameter</td>
</tr>
<tr>
<td>$d^0$</td>
<td>Coefficient in radial momentum equation</td>
</tr>
<tr>
<td>$d^1$</td>
<td>Coefficient in radial momentum equation</td>
</tr>
<tr>
<td>$d^2$</td>
<td>Coefficient in radial momentum equation</td>
</tr>
<tr>
<td>$D_{vg}$</td>
<td>Diffusion coefficient for vapor in the gas</td>
</tr>
<tr>
<td>$D$</td>
<td>Jet diameter</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>$D_H$</td>
<td>Hydraulic diameter</td>
</tr>
<tr>
<td>$f_i$</td>
<td>Coefficient of friction</td>
</tr>
<tr>
<td>$g$</td>
<td>Gravitational constant</td>
</tr>
<tr>
<td>$g_c$</td>
<td>Conversion Factor</td>
</tr>
<tr>
<td>$h$</td>
<td>Heat transfer coefficient</td>
</tr>
<tr>
<td>$h_i$</td>
<td>Heat transfer coefficient at the interface</td>
</tr>
<tr>
<td>$h_{li}$</td>
<td>Liquid side heat transfer coefficient</td>
</tr>
<tr>
<td>$h_{iv}$</td>
<td>Vapor side heat transfer coefficient</td>
</tr>
<tr>
<td>$i$</td>
<td>Enthalpy</td>
</tr>
<tr>
<td>$i_{fs}$</td>
<td>Latent heat of vaporization</td>
</tr>
<tr>
<td>$i_{lj}$</td>
<td>Enthalpy in the liquid jet</td>
</tr>
<tr>
<td>$i_{vb}$</td>
<td>Enthalpy in the vapor boundary layer</td>
</tr>
<tr>
<td>$j$</td>
<td>Mass flux</td>
</tr>
<tr>
<td>$J_v$</td>
<td>Mass removal rate from vapor phase to liquid jet model</td>
</tr>
<tr>
<td>$J_l$</td>
<td>Mass addition rate from liquid jet model to continuous liquid</td>
</tr>
<tr>
<td>$k$</td>
<td>Thermal conductivity</td>
</tr>
<tr>
<td>$k_f$</td>
<td>Thermal conductivity of liquid</td>
</tr>
<tr>
<td>$k_G$</td>
<td>Effective conductivity</td>
</tr>
<tr>
<td>$K$</td>
<td>Constant kinematic momentum</td>
</tr>
<tr>
<td>$K_I$</td>
<td>Interfacial shear coefficient</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>$k_v$</td>
<td>Thermal conductivity of the vapor/gas mixture</td>
</tr>
<tr>
<td>$l$</td>
<td>Nozzle length</td>
</tr>
<tr>
<td>$L$</td>
<td>Jet Length</td>
</tr>
<tr>
<td>$L_{tb}$</td>
<td>Thermal boundary length</td>
</tr>
<tr>
<td>$m$</td>
<td>Cooling Ratio</td>
</tr>
<tr>
<td>$M$</td>
<td>Molecular Weight</td>
</tr>
<tr>
<td>$p$</td>
<td>Pressure</td>
</tr>
<tr>
<td>$p_f$</td>
<td>Liquid Pressure</td>
</tr>
<tr>
<td>$p_g$</td>
<td>Gas Pressure</td>
</tr>
<tr>
<td>$p_v$</td>
<td>Vapor Pressure</td>
</tr>
<tr>
<td>$\dot{q}$</td>
<td>Heat rate</td>
</tr>
<tr>
<td>$\dot{q}_{li}$</td>
<td>Liquid side heat rate</td>
</tr>
<tr>
<td>$\dot{q}_{iv}$</td>
<td>Vapor side heat rate</td>
</tr>
<tr>
<td>$q^\prime$</td>
<td>Heat flux</td>
</tr>
<tr>
<td>$\dot{q}^\prime$</td>
<td>Volumetric heat rate</td>
</tr>
<tr>
<td>$q^\prime\prime$</td>
<td>Sensible heat flux</td>
</tr>
<tr>
<td>$r$</td>
<td>Coordinate in the radial direction</td>
</tr>
<tr>
<td>$R$</td>
<td>Radius</td>
</tr>
<tr>
<td>$R_{io}$</td>
<td>Jet initial radius</td>
</tr>
<tr>
<td>$R_i$</td>
<td>Radius of curvature in radial direction</td>
</tr>
<tr>
<td>$R_2$</td>
<td>Radius of curvature in axial direction</td>
</tr>
</tbody>
</table>
$R_x$ Radius of jet as a function of $x$

$R$ Universal Gas Constant

$t$ Time

$t^*$ Renewal time

$T$ Temperature

$T_{sat}$ Saturation temperature

$T_v$ Vapor Temperature

$T_L$ Liquid Temperature

$T_0$ Jet inlet temperature

$u$ Velocity

$u^0$ Axial velocity estimate in axial momentum equation

$u_f$ Liquid velocity

$u_m$ Maximum velocity

$u_v$ Vapor velocity

$u_{rel}$ Relative velocity

$v$ Radial velocity

$v^0$ Radial velocity estimate in radial momentum equation

$\vec{v}$ Velocity vector

$w$ Mass flow rate

$x$ Coordinate in the axial direction

$x_g$ Gas Fraction
Condensation heat transfer suppression factor

Greek Symbols

\( x_{\text{cond}} \)

\( \alpha \)  
Thermal Diffusivity

\( \alpha_g \)  
Noncondensable gas concentration

\( \beta_i \)  
Bessel function

\( \gamma \)  
Scalar quantity

\( \dot{\Gamma}_c \)  
Condensation Rate

\( \epsilon_D \)  
Eddy Diffusivity of Diffusion

\( \epsilon_H \)  
Eddy Diffusivity of Heat

\( \epsilon_M \)  
Eddy Diffusivity of Momentum

\( \rho \)  
Density

\( \rho_v \)  
Vapor Density

\( \rho_g \)  
Gas Density

\( \rho_f \)  
Liquid Density

\( \rho_j \)  
Liquid Jet Density

\( \rho_{vb} \)  
Vapor Boundary Layer Density

\( \phi_c \)  
Condensation Heat Flux

\( \tau_{xy} \)  
Interfacial shear stress

\( \nu_f \)  
Liquid Kinematic Viscosity

\( \nu_v \)  
Vapor Kinematic Viscosity
η Dimensionless coordinate
Δ Change
μ Liquid Viscosity
μ_v Vapor Viscosity
σ Surface Tension/Stress Tensor

Dimensionless Groups

Fr Froude Number

\[ Fr = \frac{u}{\sqrt{gL}} \]

K Kutateladze Number

\[ K = \frac{q^*}{h_{fs} \rho_v \left[ \frac{\sigma g_c g (\rho_f - \rho_v)}{\rho_v^2} \right]^{\frac{1}{4}}} \]

\( L/D \) Length to Diameter Ratio

Pe Peclet Number

\[ Pe = Re \cdot Pr = \frac{D_i u}{\alpha} \]

Pr Prandtl Number

\[ Pr = \frac{\nu}{\alpha} = \frac{\mu c_p}{k} \]

Re Reynolds Number

\[ Re = \frac{\rho u D_h}{\mu} = \frac{u D_h}{\nu} \]

St Stanton Number

\[ St = \frac{h}{\rho c_p u} \]

We_v Vapor Weber Number

\[ We_v = \frac{\rho_i u^2 D}{\sigma} \]

We_f Liquid Weber Number

\[ We_f = \frac{\rho_f u^2 D}{\sigma} \]
ACKNOWLEDGEMENTS

I would like to thank my advisors, Professor Lawrence Hochreiter and Professor John Mahaffy, for all of the time, support, and effort they provided in my tenure as a graduate student at Penn State. I would also like to thank my advisor at Bettis Atomic Power Laboratory, Dr. David Aumiller. Their expertise in thermal hydraulic code development, heat transfer and fluid flow have been invaluable during the course of the completion of this work. I would also like to acknowledge the contributions of the other members of my committee: Professor Fan-Bill Cheung and Professor Cengiz Camci. Their contributions and review of this work have resulted in a superior finished product.

I would like to acknowledge the support of my family and friends. My parents, Frank and Joan, my sister, Kerry, and my brother-in-law, Frank, have been instrumental in this work. I am also thankful for the support, understanding, inspiration, and encouragement provided by Julie Stone, which has resulted in this finished work. I would also like to thank all of my officemates and friends. Their time, companionship and conversations have helped shape this work and my time as a post-graduate student.

The financial support of the Naval Nuclear Propulsion Program is also acknowledged. I would like to thank everyone involved with the Naval Nuclear Propulsion Fellowship program for their support, without which this work would not have been possible.
Chapter 1

Introduction

The ability to model the thermal behavior of a nuclear reactor is of utmost importance to the reactor designer. It must be ensured that a nuclear reactor performs within certain thermal design limits for normal operation and all postulated transients within the design basis of the plant. Because the reactor systems are quite complicated; their thermal response cannot be calculated on a first principles basis. Large computer codes are used to calculate the system’s response to these postulated transients. COBRA-TF (COolant Boiling in Rod Arrays- Two Fluid) is one of these large codes that is used to simulate the transient behavior of the reactor vessels [1]. The topic of this work is the addition of the capability to model direct condensation on liquid jets to COBRA-TF.

Condensation is an important phenomenon when modeling a reactor system’s response to a Loss Of Coolant Accident (LOCA). Condensation is even more important with the use of passive safety systems which rely on condensation heat transfer for long term cooling. Condensation analysis has typically been focused on liquid layers forming a thin film on a cold wall, however in reactor applications condensation can also occur on turbulent jets or sprays. Sprays may be found in the pressure suppression systems in the reactor containment as well as other locations. Jets may be formed when Emergency Core Cooling (ECCS) water enters the large coolant piping from smaller piping [2]. The models used in COBRA-TF to predict the suppression of condensation heat transfer in
the presence of noncondensable gases were developed from jet condensation experiments. [3] The condensation suppression model currently used in COBRA-TF is similar to that used in the TRAC code, which was derived from work by Sklover and Rodivilin.[4]

The purpose of this work is to improve the COBRA-TF capability to model direct jet condensation heat transfer over the range of conditions in reactor transients which create turbulent jets using an embedded liquid jet model. In order to accomplish this task, a stand-alone liquid jet model with a separate set of equations of motion and energy were developed for a liquid jet and the surrounding gas boundary layer. In addition, the required constitutive relationships were developed such that the stand-alone jet model could be coupled to COBRA-TF.

COBRA-TF is used in a subchannel mode for the coupling of the stand-alone jet model. In the subchannel analysis mode, it is assumed that the flow in the axial direction is dominate. Simplifications are made to the lateral momentum equations in which donor cell momentums are assumed to be lost when flow goes through a gap. The assumptions in the lateral momentum and the use of separate transverse momentum cells allow the use of simplified equations for which no specific lateral coordinate is needed. Separate momentum control volumes are used in the axial and lateral directions as shown in Figure 1-1 below. The subchannel mode is not fully three dimensional because of the assumption made in the transverse momentum equations which eliminate the lateral
transport of lateral momentum. However, in the subchannel mode, the control volumes can combined in an arbitrary way to simulate a three-dimensional flow field.

Like most large thermal-hydraulic system codes, COBRA-TF divides the reactor into many volumes. The temperature, pressure, velocity, and void fraction in each volume is found by simultaneously solving the conservation equations and applying the equation of state. Unlike most thermal-hydraulic system codes which model vapor and liquid fields, COBRA-TF models three fields for two-fluid flow. It contains models for a vapor field with noncondensable gas modeling, a continuous liquid field, and an entrained droplet field.

COBRA-TF also models the transport of a noncondensable gas mixture with the vapor field and solves a total of nine conservation equations for the three fields. Included
in the nine conservation equations are four mass equations, two energy equations, and three vector momentum equations. There are mass equations for vapor, noncondensable gases, continuous liquid, and entrained droplets. The two energy equations are a combined liquid equation and a combined vapor-gas equation. This requires that the continuous liquid and entrained droplet fields have the same temperature in a given computational cell. Vector momentum equations exist for the combined vapor-gas, continuous liquid, and entrained droplets. In COBRA-TF, the continuous liquid field is modeled as a liquid film. In this approach, the liquid film must be in contact with a wall and all computational cells must have a non-zero hydraulic diameter. This stipulation does not lend itself to modeling of a liquid jet accurately.

The proposed stand-alone liquid jet model is designed to model the liquid jet behavior more mechanistically. In order to do this, Eulerian liquid jet mass, momentum, and energy equations have been developed to model explicitly the liquid jet. The stand-alone jet model is then added to COBRA-TF using a secondary mesh. The stand-alone jet equations are required since the jet temperature is can be different as than other liquid fields at the same location. The stand-alone jet conservation equations will interface with the existing COBRA-TF fields equations as mass, momentum, and energy sources and sinks. A finer volume discretization will be used in coordination with the new stand-alone jet model in an attempt to more accurately model the temperature distribution within the jet and the heat and mass transfer occurring at the interface between the jet and its surrounding fluid. The sub-grid will contain the liquid jet and the boundary region in the vapor surrounding the jet to the point at which the axial component of the velocity of
the vapor is zero. Figure 1-2 shows the streamlines that would be associated with the vapor boundary layer flow.

This will allow the capture of the entire region of interest for problems with condensing jets within the new model as well as simplify the interface of the new model with existing thermal-hydraulic system codes. In order to interface between the primary and secondary mesh, the secondary stand-alone jet model mesh is limited to exist within a single COBRA-TF channel.
1.1 Problem Description

The problem that is of interest for this work is that of a sub-cooled liquid jet being injected into a quiescent steam or steam and noncondensable gas mixture. The mixture that the jet is being injected into is maintained at or above the saturation temperature and at the local steam partial pressure. A physical picture of a liquid jet being injected into a quiescent vapor environment is shown in Figure 1-3. There are two cases which will be discussed in this description; the first is the case in which the jet is injected into an environment of pure steam and in the second case the steam will be mixed with a noncondensable gas.
In the first case there is no impediment for condensation of the steam on the subcooled water jet. As the jet enters the steam environment two related boundary layers will form. The entering jet will induce a velocity within the vapor around the jet due to the interfacial drag between the jet and vapor. If the vapor is superheated, a thermal gradient will also exist in the vapor surrounding the liquid jet. Within the jet itself, both velocity and temperature gradients will form.
The jet, immediately before coming into contact with the steam, is assumed to have uniform velocity and temperature. As the jet makes contact with vapor, condensation will begin to occur. The condensate will form at the interface on the outside of the jet bringing with it mass, momentum, and energy. This condensate will be at the saturation temperature and have a very small velocity (that of the interface) when initially deposited. A velocity gradient will develop within the jet with the interface velocity being the local velocity at the edge of the jet and some maximum velocity in the center of the jet. Due to turbulent mixing and conduction within the jet a thermal boundary layer will also form in the jet. The core of the jet will remain sub-cooled while the edge of the jet will be at the saturation temperature. The turbulent nature of the liquid jet will most likely result in similar velocity and temperature boundary layer profiles according to heat and momentum turbulent eddy diffusivity similarity when the Prandtl Number is near one. The Prandtl number represents the ratio between heat and momentum diffusivity. If the Prandtl number is near unity the thermal and momentum boundary layers will also be of the same thickness.

At the boundary between the liquid jet and the steam; there will exist an interface temperature and velocity. The interface temperature will be the saturation temperature. An interface velocity will be induced by the motion of the jet through the surrounding vapor. The interface velocity will have to be determined by simultaneously solving the boundary layer equations in the liquid jet and the vapor. The location of the interface will need to be determined by solving the jet mass equation for the radius of the jet as a
function of time and the axial location. The radius of the jet will change with the addition of new mass from condensation as well as the dynamics of the jet flow.

In addition to the axial velocity induced in the vapor, there will be a net flow of vapor towards the jet in the transverse direction because of the condensation occurring at the interface with the jet. This steam velocity will be proportional to the condensation heat transfer rate. The size of the vapor boundary layer must also be determined. At the outer edge of the vapor boundary layer, determined by the induced axial velocity, the only velocity will be in the transverse direction.

The only difference between the first case, without noncondensables, and second case, with noncondensables, occurs in the gas boundary layer. In the second case noncondensible gases are introduced that act to retard the condensation heat transfer process by providing resistance to the mass transfer of the vapor at the jet. Figure 1-4 shows a liquid/vapor interface with gas accumulation. When the vapor/gas mixture flows to the jet, the gas is left behind at the surface of the jet when the vapor condenses. This noncondensible gas will then form a layer through which the vapor must diffuse in order to reach the jet surface to condense. This noncondensible gas layer can greatly reduce the condensation heat transfer rate at the surface of the jet. The buildup and sustainability of the gas layer can be disrupted by the turbulent nature of the gas/liquid interface. Turbulence at the interface disrupts the existing layer of gas and renews the bare interface between the jet and the vapor/gas mixture. When this occurs the gas that has built up at the interface will mix into the bulk vapor/gas surrounding the jet. The time allowed for a
noncondensable gas layer to build at the interface is called the renewal time.[2] In the case of a turbulent liquid jet the gas layer the renewal time of the gas layer is generally small.

A small renewal time will decrease the ability of the noncondensable gas to hinder the condensation rate. Reducing the renewal time for gas buildup both decreases the thickness of the gas layer that is permitted to form and limits the time the gas is present at the interface.
1.2 Originality of Current Work

The current work contains several original concepts. This work is unique in its modeling of a continuous liquid field in a reactor vessel analysis code as a jet and not as a film. The constitutive relationships needed for modeling the liquid jet are unique in a reactor vessel analysis code. The use of an overset grid method for a secondary mesh is required and is also unique for this type of simulation. Overset grid methods have been used for other purposes in subchannel codes where a finer volume discretization is beneficial. These methods have been used when modeling the quench front in reflood heat transfer [1]. In reactor vessel analysis codes, for a given volume the temperature and velocity of the fluid field are solved as average values. In this work, the local temperature and velocity gradients are solved within the liquid jet. Also, the jet model will simulate the boundary layer in the vapor or vapor and gas mixture surrounding the jet. Solving for the detailed boundary layer behavior at the jet interface, as well as the local jet liquid temperatures will permit more accurate calculations of the condensation heat flux rate at the surface of the liquid jet.
Chapter 2

Experiments on Condensation Heat Transfer on Liquid Jets

Experiments involving condensation on liquid jets have been motivated by many different industries for many years. In the early 1950’s experiments were performed for direct condenser, direct heaters, and thermal deaerators. More recently work has been inspired by emergency core cooling systems in the nuclear industry. Overall, the number of quality papers investigating jet condensation is quite small. However, a search of the literature uncovers several studies involving condensing jets, both with and without noncondensable gases present. Most of the experiments consist of a liquid jet exiting a nozzle and traveling downward in a vapor or vapor and gas environment. A jet collection and temperature measuring device is placed a certain distance below the nozzle to measure the change in the average temperature of the jet. Figure 2-1 shows an example of the type of apparatus used for a jet condensation experiment [19]. The nozzle at the top of the test section injects water vertically downward into the condensing environment. A collection and mixing device is located at the bottom of the test section so that the average temperature of the jet can be recorded. In most of the experiments found, the distance between the nozzle and the downstream temperature measurement is fixed during any given test.
Sklover and Rodivilin [39] performed a series of tests with fifty separate jets in a rectangular array with varying pitch between the jets from 10-25mm, with and without noncondensable gases present. From this test they determined the relationship for condensation suppression in the presence of noncondensable gases given in Eq. 2-1 as

$$x_{\text{cond}} = 0.336 (\epsilon m)^{-0.2}$$  \hspace{1cm} (2-1)
where $\varepsilon$ is the noncondensable gas fraction and $m$ is the cooling ratio, which is defined as the ratio of the mass flux of liquid to the mass flux of vapor. The condensation suppression factor, given as $x_{\text{cond}}$, is defined as the ratio of the condensation rate with noncondensables present to the condensation rate at the same conditions without noncondensables. This is shown as Eq. 2.2 where $T_{v,i}$ is the vapor temperature at the interface, which is equal to the saturation temperature at the steam partial pressure at the interface, $T_{v,0}$ is the bulk vapor temperature, and $T_f$ is the temperature of the liquid.

$$x_{\text{cond}} = \frac{h_l \left( T_{v,i} - T_f \right)}{h_l \left( T_{v,0} - T_f \right)} = \frac{T_{v,i} - T_f}{T_{v,0} - T_f}$$ (2.2)

There are several experiments which investigate the effects of noncondensable gases on a single jet as described above. Sklover and Rodivilin [4] performed a set of experiments with relatively large diameter jets (between 15 and 30 mm) and L/D ranging from twelve to sixty. For each experiment one measurement was taken at a fixed distance from the inlet nozzle. A correlation for the Stanton number was derived as

$$St = 2.7 \text{Re}^{-0.4} \text{Pr}^{-0.55} K^{-0.11} \text{We}_v^{0.4} \left( \frac{L}{D} \right)^{-0.6}$$ (2.3)

where $\text{Re}$ is the Reynolds number, $\text{Pr}$ is the Prandtl number, $K$ is the Kutateladze number, $\text{We}_v$ is the vapor Weber number and L/D is the ratio of the jet length to its initial diameter. The condensation suppression factor in the presence of noncondensable gases was developed from this experimental data as

$$x_{\text{cond}} = 0.168 (\varepsilon m)^{-0.1} \left( \frac{L}{D} \right)^{0.33}$$ (2.4)
In Eq. 2-4 $\varepsilon$ is the concentration of gas in the vapor, $m$ is the cooling ratio, which is defined as the ratio of the mass flux of liquid to the mass flux of vapor, and $L/D$ is the ratio of the jet length to the jet initial diameter. In the cooling ratio, the mass flux of liquid is the mass flux of the jet and the mass flux of the vapor is proportional to the condensation heat transfer rate.

Another test involving a single jet injected into an environment with noncondensable gases present was the set of experiments by Isachenko and Solodov [17]. In this test the jet diameters were varied between 2.2-6 mm with $L/D$ between 4 and 180. The data is divided into two regions and separate correlations for the Stanton number have been obtained. The two regions are determined from the vapor Weber number; which uses the vapor density, relative velocity between the vapor and liquid jet, the jet diameter, and the surface tension. If the Weber number is above 2.7 Eq. 2-5 is used and if the Weber number is below 2.7 Eq. 2-6 is used.

For $We_v > 2.7$

$$St = 0.00335 \text{Re}^{-0.17} \text{Pr}^{-0.09} We^{0.35} K^{0.13} \left( \frac{L}{D} \right)^{-0.42}$$  \hspace{1cm} (2-5)$$

For $We_v < 2.7$

$$St = 0.00333 \text{Re}^{-0.18} \text{Pr}^{-0.05} K^{0.11} \left( \frac{L}{D} \right)^{-0.41} \exp[0.16 We_v^{0.35}]$$  \hspace{1cm} (2-6)$$

Several jet condensation experiments in which the jets are injected into pure steam are also relevant to this study. Takahashi et al [19], Trofimov [40], Isachenko et al [16], and Sklover and Rodivilin [41] all performed experiments with jets of various
lengths. However it is not apparent whether the measurements were made during a single steady state experiment or in separate tests under similar circumstances.

The tests by Celata et al [12] had a single jet length with temperature measurements that were made along the entire length of the jet during a single steady state experiment. For this reason, this test is ideal for comparison with the stand-alone liquid jet model for this work. The range of data for this experiment is given in Table 2-1. Ten measurements of the jet temperature were taken along the length of the jet during an average experiment.

Table 2-1: Range of parameters for Celata test

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Saturated steam temperature (°C)</td>
<td>105 – 155</td>
</tr>
<tr>
<td>Inlet liquid temperature (°C)</td>
<td>15 – 80</td>
</tr>
<tr>
<td>Liquid mass flow rate (g/s)</td>
<td>1.5 – 25</td>
</tr>
<tr>
<td>Inlet mean liquid velocity (m/s)</td>
<td>0.08 – 20</td>
</tr>
<tr>
<td>Jet nozzle diameter (mm)</td>
<td>1, 2, 3, 5</td>
</tr>
<tr>
<td>Jet nozzle length (diameters)</td>
<td>1 and 20</td>
</tr>
<tr>
<td>Jet length (mm)</td>
<td>250</td>
</tr>
</tbody>
</table>

In Celata et al’s experiment, the saturated steam is produced in a 15 kW electric boiler which can produce 20 kg/hr. A 10 kW electric heater is used to regulate the liquid inlet temperature which is measured with a thermocouple upstream of the nozzle. The downstream average temperature of the liquid jet is measured using a thermocouple in a small mixing cup device. The water mass flow rate is measured using turbine flowmeters. For each test run, the nozzle L/D, nozzle diameter, saturated steam temperature, liquid flow rate, and several measurements of the jet temperature.
downstream of the injection point along with the distance from the nozzle the
measurement were obtained. A complete listing of the test data is given in Appendix 1.

Four sources of uncertainty were identified by the experimenters [12]. There are
uncertainties associated with the evaluation of the distance between the nozzle outlet and
the location of the temperature measurement, with the instability of the liquid jet, with
oscillations of steam pressure inside the test vessel, and with the inlet water temperature
measurement. At low liquid flow rates the water can be heated by the nozzle walls. A
histogram was developed by Celata et al to characterize the uncertainty in the
experiments and is shown in Figure 2-2 below. The horizontal axis shows the uncertainty
and the vertical axis shows the rate at which that uncertainty occurs in the data.

Figure 2-2: Uncertainty in Celata Jet Condensation Test
Chapter 3
Development of the Stand-alone Jet Computational Model

The computational model for the liquid jet and the surrounding vapor boundary layer has been developed as a stand alone module that can be added to the COBRA-TF analysis code. Using this approach, the newly developed model and its stand alone implementation can be easily adapted to compliment other code packages.

The current work includes models for a continuous liquid jet as well as the vapor boundary layer surrounding the jet. The vapor boundary layer surrounding the jet will be defined by the axial velocity induced by the liquid jet. The boundary begins at the interface and extends to a radius R, which is given at $u_z(R)=0$, at which the vapor velocity in the axial direction is equal to zero. This can be seen in Figure 1-2 as the location where the streamlines are perpendicular to the jet. This boundary region is of special interest when noncondensible gases are present in the vapor. The size of the boundary region will change with time and along the length of the jet. The thickness of the vapor-gas region to be solved is input by the user and should be large enough to include the entire vapor boundary layer.
3.1 Background for Equations of Motion for Liquid Jet

Problems involving a liquid jet will be solved using a coupled Eulerian-Eulerian approach. The liquid jet and the surrounding vapor boundary layer will be solved using Eulerian equations of motion and be coupled to the standard Eulerian solution used in COBRA-TF. The motion of the liquid jet will be represented by four separate equations. These will be independent mass, axial and radial momentum and energy equations. These equations will be solved semi-implicitly. An estimated velocity and its change with pressure and enthalpy is found using old time conditions. The continuity and energy are then solved, for the new time pressure and enthalpy, using new time velocities, which are functions of the new time pressures and enthalpies, and old time pressures and enthalpies. The equation of state is then solved for new time densities and other state variables. Finally the new time velocities are found using the new time pressure and enthalpy. In the vapor boundary layer an additional equation will be solved. A non-condensable gas concentration equation will also be solved simultaneously with the continuity and energy equations. The results of the Eulerian jet calculation will then be used as vapor and liquid mass and energy sources and sinks for the solution of the COBRA-TF calculation.

The four liquid jet equations of motion and energy will be of the similar form as the equations of motion and energy for the continuous liquid phase currently used in COBRA-TF. The starting point for the development of a liquid jet model comes from Kutateladze [7]. In the following formulations, $x$ is given as the axial coordinate in the
solution of the jet equations. A steady-state liquid jet momentum equation, taken from Kutateladze [7] and using his nomenclature, is given in Equation 3-1 as

\[
g \rho_f \frac{dp}{dx} = \rho_f u_f \frac{du_f}{dx} + \frac{u_f - u_v}{\pi R_x^2 g} \Pi_c \frac{d\Gamma_c}{dx} \tag{3-1}
\]

In the above equation, the pressure in the jet is related to the vapor pressure by Equation 3-2, as

\[
p_f = p_g + \sigma \left( \frac{1}{R_1} + \frac{1}{R_2} \right) \tag{3-2}
\]

In Equation 3-2, \( R_1 \) and \( R_2 \) are the radii of curvature of the jet in the axial and radial directions. The radius of curvature in the axial direction of the jet is much larger than the radius of jet so it can be assumed that:

\[
\frac{1}{R_1} + \frac{1}{R_2} = \frac{1}{R_x} \tag{3-3}
\]

Assuming that the vapor pressure and the surface tension only change with gravity in the \( x \) direction:

\[
\frac{dp}{dx} = g \rho_g + \frac{d}{dx} \left( p_v + \frac{\sigma}{R_x} \right) = g \rho_g - \frac{\sigma}{R_x^2} \frac{dR_x}{dx} \tag{3-4}
\]

The equation of motion of the jet becomes:

\[
\left( \rho_f - \rho_g \right) g + \frac{\sigma}{R_x^2} \frac{dR_x}{dx} = \rho_f u_f \frac{du_f}{dx} + \frac{u_f - u_g}{\pi R_x^2 g} \Pi_c \frac{d\Gamma_c}{dx} \tag{3-5}
\]

The liquid jet continuity equation from Kutateladze is given in Equation 3-6 as

\[
\pi R_x^2 \rho_f u_f = \pi R_0^2 \rho_f u_{f_0} + \int_0^x d\Gamma_c \tag{3-6}
\]
where:

$$i g \frac{d \Gamma_c}{dx} = \pi R_x^2 u_f c_p \rho_f \frac{dT_f}{dx} \quad (3-7)$$

The liquid jet energy equation is given as:

$$w_x \frac{\partial T}{\partial x} = \frac{k + k_T}{c_p \rho_f g} \left( \frac{\partial^2 T}{\partial r^2} + \frac{1}{R} \frac{\partial T}{\partial R} \right) \quad (3-8)$$

These equations formed the framework for the investigation into liquid jet flows. The one dimensional nature of the above set of equations from Kutateladze limit their ability to predict the radial temperature profile within the liquid jet. Being able to characterize the temperature profile enables the ability to more accurately predict the heat transfer between the liquid jet and its surroundings at the surface of the jet with the vapor boundary layer. A two dimensional approach would allow this more detailed information to be simulated. A two dimensional simulation of the vapor boundary layer will also allow more accurate predictions of the vapor temperature and non-condensable gas distributions. For the stand-alone jet model, two dimensional flow equations were developed for the jet behavior and are discussed below.

### 3.2 Equations of Motion for Liquid Jet

There are four equations that are solved to simulate the liquid jet. These equations are: the liquid continuity equation, the liquid energy equation, and the jet axial and radial momentum equations. The liquid jet flow equations are two dimensional,
conservative, and compressible. The general form of these equations are developed below.

The general form of the liquid jet continuity equation is given as

\[
\frac{\partial \rho_{lj}}{\partial t} + \nabla \cdot (\rho_{lj} \vec{v}_{lj}) = \dot{\Gamma}\omega
\]  \hspace{1cm} (3.9)

In the above equation, the transport of mass is modeled using the time rate change of liquid jet density, \( \rho_{lj} \). The density is convected by the velocity vector, \( \vec{v}_{lj} \), on the left hand side of the equation and on the right hand side of the equation is the mass source due to mass transfer, \( \dot{\Gamma}\omega \).

The general form of the liquid jet energy equation is given as

\[
\frac{\partial \rho_{lj} i_{lj}}{\partial t} + \nabla \cdot (\rho_{lj} i_{lj} \vec{v}_{lj}) = \dot{q}_{lj} - \dot{\Gamma}\omega i_{lj} + \frac{\partial p}{\partial t} + \epsilon_{Mj} \nabla^2 \rho_{lj} i_{lj}
\]  \hspace{1cm} (3.10)

In the above equation, \( \rho_{lj} \) is the liquid jet density, \( i_{lj} \) is the liquid jet enthalpy, \( \vec{v}_{lj} \) is the liquid jet velocity vector, \( \dot{q}_{lj} \) is the liquid side volumetric heat rate, \( \dot{\Gamma}\omega \) is the mass transfer per unit volume, and \( \epsilon_{Mj} \) is the momentum diffusivity. In Equation 3.10 the time rate of change of energy and the convection of energy by the velocity are equal to the energy source from mass transfer, the pressure work, and the diffusion, either molecular or turbulent, of energy.
The liquid jet axial momentum equation is given as

$$\frac{\partial \rho_j u_j}{\partial t} + \nabla \cdot \left( \rho_j u_j v^j \right) = \Gamma^\sigma u_j + \rho_j g + \frac{\partial p}{\partial x} + \varepsilon_M \nabla^2 \rho_j u_j \quad (3.11)$$

In the above equation, \(u_j\) is the liquid jet axial velocity, \(g\) is the acceleration due to gravity, \(\varepsilon_M\) is the liquid jet momentum diffusivity. The time rate change of axial momentum and the convection of axial momentum by the velocity vector appear on the left hand side of Equation 3.11. Appearing on the right hand side are the momentum source due to mass transfer, body force, pressure force, and diffusion of axial momentum terms.

The liquid jet radial momentum equation is given as

$$\frac{\partial \rho_j v_j}{\partial t} + \nabla \cdot \left( \rho_j v_j v^j \right) = \Gamma^\sigma v_j + \frac{\partial p}{\partial r} + \varepsilon_M \nabla^2 \rho_j v_j \quad (3.12)$$

In the radial momentum equation \(v_j\) is the liquid jet radial velocity. The radial momentum equation contains the same terms as the axial momentum equation with the exception of the gravitation force term. The vapor boundary layer contains similar flow equations which are given in the next section.

3.3 Equations of Motion for Vapor Boundary

Five equations will be solved to simulate the vapor boundary layer. These equations include: the vapor continuity equation, axial and radial momentum equations, vapor energy equation, and a non-condensable gas concentration equation. The equations
for the vapor boundary layer are similar to the flow equations used in COBRA-TF.

However in the current work the flow equations are truly two dimensional and
conservative whereas the COBRA-TF transverse flow terms are not fully conservative.

The general form of the vapor continuity equation is given as

\[
\frac{\partial \rho_{vb}}{\partial t} + \nabla \cdot (\rho_{vb} \vec{v}_{vb}) = \Gamma^{\infty}
\]  

(3.13)

In the above equation the time rate change of vapor boundary layer density, \( \rho_{vb} \), is
related to the efflux of mass by the vapor velocity vector, \( \vec{v}_{vb} \), and mass transfer, \( \Gamma^{\infty} \).

For this first implementation of the vapor boundary layer equations, it is assumed that the
noncondensable gas density is equal to the vapor density, and both are equal to the vapor
boundary layer density.

The general form of the energy equation is given in Equation 3.14 as

\[
\frac{\partial \rho_{vb} i_{vb}}{\partial t} + \nabla \cdot (\rho_{vb} i_{vb} \vec{v}_{vb}) = \Gamma^{\infty} \dot{i}_{vb} + \frac{\partial p}{\partial t} + \varepsilon_H \nabla^2 \rho_{vb} i_{vb}
\]  

(3.14)

The transport of energy, in the form of vapor enthalpy, \( i_{vb} \), is modeled in the above
equation as the efflux of energy by the vapor velocity vector, the addition or subtraction
of energy through mass transfer, pressure work, and thermal diffusion, molecular and/or
turbulent. The thermal diffusivity, either molecular or turbulent, is represented by \( \varepsilon_H \).

The non-condensable gas transport equation is given in Equation 3.15 as
In the above equation \( \alpha_g \) is the non-condensable gas concentration. The turbulent and/or molecular diffusivity of the non-condensable gas in vapor is given as \( \varepsilon_D \). In Equation 3.15 the rate change in non-condensable gas concentration and the convection of the gas concentration by the vapor velocity vector are on the left hand side and the diffusion of the gas is on the right hand side.

The axial momentum equation is given in Equation 3.16 as

\[
\frac{\partial \rho_{ib} v_{ib}}{\partial t} + \nabla \cdot \left( \rho_{ib} v_{ib} \bar{v}_{ib} \right) = \dot{\Gamma}^\rho \bar{v}_{ib} + \rho_{ib} g + \frac{\partial p}{\partial x} + \varepsilon_M \nabla^2 \rho_{ib} v_{ib}
\]

(3.16)

The axial velocity, \( u \), is solved for using the axial momentum equation above. The time rate change of axial momentum and the convection of the axial momentum by the velocity vector, \( \bar{v} \), are on the left hand side of the equation. On the right hand side are the momentum source from mass transfer, body force term, pressure force term, and momentum diffusivity term with a momentum diffusivity given by \( \varepsilon_M \).

The radial momentum equation is given in Equation 3.17 as

\[
\frac{\partial \rho_{ib} v_{ib}}{\partial t} + \nabla \cdot \left( \rho_{ib} v_{ib} \bar{v}_{ib} \right) = \dot{\Gamma}^\rho \bar{v}_{ib} + \frac{\partial p}{\partial r} + \varepsilon_M \nabla^2 \rho_{ib} v_{ib}
\]

(3.17)

The transport of the radial velocity, \( v \), is described by the radial momentum equation. The time rate change of radial momentum is determined by the convection of the radial momentum by the velocity vector, \( \bar{v} \), on the left hand side of the above equation and the
radial momentum source from mass transfer, the radial pressure work, and the diffusion of radial momentum on the left hand side.

The linearization of the equations describing both the liquid jet and the vapor boundary layer is described in Chapter 4.

3.4 Required Constitutive Relationships

A number of constitutive relationships are needed in order to close the above equations. The required relationships are: jet breakup, interfacial heat transfer, sensible heat transfer, turbulence in the jet, effects of noncondensable gas on condensation, jet formation, and jet destruction and interface with continuous liquid field.

3.4.1 Jet Breakup

There are four basic regimes of jet breakup. In both turbulent and laminar jets, both small and large droplets can form. The physical forces required to determine the regime of jet breakup are surface tension, inertial forces in the fluids, viscous forces, and the body force. These forces are characterized by Reynolds number, Weber number, Froude number, density ratios, and the velocities of the fluids involved. The process of liquid jet breakup is prohibitively complex [42,43] for use in this jet model and for that reason is considered to be beyond the scope of this work. It is assumed that the jet is continuous over its entire length for the jets of interest in this thesis.
3.4.2 Interfacial Heat Transfer

The interfacial heat transfer coefficient is defined by Collier [8] as

$$h_i = \frac{\phi}{\Delta T}$$ \hspace{1cm} (3-18)

In Equation 3-18 \(\phi\) is the condensation heat flux, which is defined as the product of the latent heat of vaporization, \(i_{fg}\), and the net mass flux across the interface, \(j\).

$$\phi_c = i_{fg} j$$ \hspace{1cm} (3-19)

The net mass flux across the interface is given by Collier and Thome to be

$$j = \left[ \frac{M}{2\pi RT} \right]^{1/2} \frac{\Delta p}{\Delta T}$$ \hspace{1cm} (3-20)

In Equation 3-20 the pressure difference is between the steam pressure and the saturation pressure at the liquid temperature. \(M\) is the molecular weight and \(R\) is the universal gas constant. Combining Equation 3-19 and Equation 3-20 yields Equation 3-21.

$$h_i = i_{fg} \left[ \frac{M}{2\pi RT} \right]^{1/2} \frac{\Delta p}{\Delta T}$$ \hspace{1cm} (3-21)

Collier and Thome also point out that the condensation rate is dependent on the surface temperature on the liquid side. Temperature difference between the vapor and liquid is the driving force for heat transfer, as this decreases, the condensation rate will decrease from the maximum rate in Equation 3-21. Collier proposes a reduced condensation rate which is a function of the thermal conductivity of the fluid, \(k_f\), thermal diffusivity of the fluid, \(\alpha_f\), and the time since condensation began, \(t\), which is given as

$$h_i = k_f \left( \frac{\pi \alpha_f t}{3} \right)^{-1/2}$$ \hspace{1cm} (3-22)
The turbulent nature of condensation on a liquid jet will most likely lead to a condensation rate which is larger than this reduced rate. The temperature gradient inside the liquid jet should be used to determine the actual interfacial heat transfer.

Celata et al [12] suggest an expression for the heat transfer coefficient for a condensing liquid jet assuming a Bessel-type temperature profile within the jet, using Celata’s nomenclature, as

\[
h_l(x) = k_G(x) \frac{\beta_1^2}{2R_0} \left(1 + \frac{2gx}{u_0^2}\right)^{1/4}
\] (3-23)

In Equation 3-23, \(\beta_1\) is the root of the Bessel function, \(R_0\) is initial radius of the jet, \(g\) is the acceleration due to gravity, \(x\) is the distance from the nozzle, and \(u_0\) is the velocity of the jet at the nozzle exit. The parameter \(k_G\) in Equation 3-23 is defined as the equivalent thermal conductivity within the jet and given as

\[
k_G(x) = k \left[1 + A(B + C)\right]
\]

where

\[
A = \left(\frac{\sqrt{D}}{D}\right) Pe^{(D/l)^{0.12}}
\]

\[
B = 0.00136 \left(\frac{L_{th}}{x}\right)^{1.38} \left(\alpha_L D\right) \exp \left[-\left(\frac{L_{th}}{x}\right) \left(\frac{l}{D}\right)\right]
\]

\[
C = 0.00036 \frac{T_s - T_j}{T_s - T_w}
\] (3-24)

In Equation 3-24 \(l\) is the length of the nozzle, \(D\) is the diameter of the nozzle \(L_{th}\) is the thermal breakup length, \(Pe\) is the Peclet number, \(\alpha_L\) is the Laplace constant, \(T_j\) is the average temperature of the jet at \(x\). The thermal breakup length, \(L_{th}\), is a parameter.
which describes the distance needed to transition from a flat temperature profile in the jet to the Bessel-type profile.

Several other empirical correlations have been developed by researchers studying liquid jets using Stanton number correlations. Most of the correlations reported are empirically derived from experimental data. Several of the investigators separated the results into two or more regions and developed separate correlations for each of these regions. Others simply made a single correlation to cover the entire range of data available. The Stanton number correlations that were developed to cover the entire range of data appear in Table 3-1 below. Those that were developed for multiple regions and the conditions for those regions appear in Table 3-2.

Table 3-1: Correlations for Stanton Number from Previous Works

<table>
<thead>
<tr>
<th>Authors</th>
<th>Reference</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vasilev (from Kutateladze)</td>
<td>[7]</td>
<td>( St = 0.01667 Fr^{-0.2} \left( \frac{L}{D} \right)^{-0.3} )</td>
</tr>
<tr>
<td>Sklover &amp; Rodivilin</td>
<td>[4]</td>
<td>( St = 2.7 \sqrt{Re}^{-0.4} Pr^{-0.55} K^{0.11} We^{0.4} \left( \frac{L}{D} \right)^{-0.6} )</td>
</tr>
<tr>
<td>Benedek</td>
<td>[13]</td>
<td>( St = 0.00263 K^{0.084} \left( \frac{L}{D} \right)^{-0.06} )</td>
</tr>
<tr>
<td>De Salve et al</td>
<td>[14]</td>
<td>( St = 3.25 Re^{-0.38} Pr^{-0.52} K^{-0.19} \left( \frac{L}{D} \right)^{-0.52} )</td>
</tr>
<tr>
<td>Kim &amp; Mills</td>
<td>[15]</td>
<td>( St = 3.2 Re^{-0.20} Pr^{-0.7} L^{0.18} Su^{-0.19} \left( \frac{L}{D} \right)^{-0.57} )</td>
</tr>
</tbody>
</table>

In Table 3-2, the correlations that were developed by Takahashi et al are separated into Regions which are defined by certain criteria. Regions A and B the effects of surface
disturbance is a dominant factor. These regions are described as having

\[ u_f \left( \frac{\mu_f}{\sigma_f} \right) \left( \frac{\rho_f}{\rho_f} \right)^{1/2} \geq 3.7 \times 10^{-5} \]. Region A is defined as having \( L/D \geq 4.8W_e f^{0.32} \) while Region B has \( L/D < 4.8W_e f^{0.32} \). The liquid Weber number is defined as

\[ W_e f = \frac{\rho_f u^2 D}{\sigma} \]  \hspace{1cm} (3.25)

The other two regions, Regions C and D, where the effects of the surface disturbances are not dominate, are defined as having \( u_f \left( \frac{\mu_f}{\sigma_f} \right) \left( \frac{\rho_f}{\rho_f} \right)^{1/2} < 3.7 \times 10^{-5} \). Region C is defined as having \( L/D \geq 4.8W_e f^{0.32} \) while Region D has \( L/D < 4.8W_e f^{0.32} \).

Table 3-2: Correlations for Stanton number with Multiple Regions

<table>
<thead>
<tr>
<th>Author</th>
<th>Reference</th>
<th>Region</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isachenko &amp; Solodov</td>
<td>[16]</td>
<td>( (L/D) \leq 95 )</td>
<td>( St = 0.0129 \left( \frac{L}{D} \right)^{-0.54} \exp[1.35W_e v] )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( (L/D) &gt; 95 )</td>
<td>( St = 0.00375 \left( \frac{L}{D} \right)^{-0.27} \exp[1.35W_e v] )</td>
</tr>
<tr>
<td>Isachenko &amp; Solodov</td>
<td>[17]</td>
<td>( W_e v &gt; 2.7 )</td>
<td>( St = 0.00335 \Re^{-0.17} \Pr^{-0.09} W_e v^{0.35} K^{0.13} \left( \frac{L}{D} \right)^{-0.42} \exp[0.16W_e v^{0.35}] )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( W_e v &lt; 2.7 )</td>
<td>( St = 0.00335 \Re^{-0.18} \Pr^{-0.05} K^{0.11} \left( \frac{L}{D} \right)^{-0.41} \exp[0.16W_e v^{0.35}] )</td>
</tr>
<tr>
<td>Iciek</td>
<td>[18]</td>
<td>( L \leq 11.5W_e l^{0.21} )</td>
<td>( St = 0.00268 \left( \frac{L}{D} \right)^{-0.27} W_e v^{0.35} Fr^{-0.07} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( L \geq 11.5W_e l^{0.31} )</td>
<td>( St = 0.0038 \left( \frac{L}{D} \right)^{-0.22} Fr^{-0.18} )</td>
</tr>
<tr>
<td>Takahashi et al</td>
<td>[19]</td>
<td>Region A</td>
<td>( St \propto \left( \frac{L}{D} \right)^{0.16} d^{1.0} \Re^{0.84} \Pr^{0.22} W_e v^{0.32} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Region B</td>
<td>( St \propto \left( \frac{L}{D} \right)^{-0.31} d^{0.86} \Re^{-0.09} \Pr^{-0.62} W_e v^{0.32} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Region C</td>
<td>( St \propto \left( \frac{L}{D} \right)^{-0.72} d^{0.90} \Re^{-0.84} \Pr^{-1.3} W_e v^{0.32} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Region D</td>
<td>( St \propto \left( \frac{L}{D} \right)^{0.40} d^{1.68} \Re^{-0.94} \Pr^{1.4} W_e v^{0.32} )</td>
</tr>
</tbody>
</table>
In the cases in Table 3-2, where the data is divided in several regions, the data is divided based primarily on the continuity of the jet surface. At longer L/D or small vapor Weber numbers the jet is expected to begin to breakup or show signs of disturbance some distance from the jet nozzle.

A simpler correlation for the heat transfer coefficient is suggested by Theofanous [45] for a turbulent jet after which is given in Equation 3.26 below.

\[
h_t = 0.02 \rho c_p u \left( \frac{x}{D} \right)^{-0.5}
\]

Equation 3.26

Theofanous developed this correlation after evaluating a large number of jet condensation experiments, including many of those listed above in Table 3-1 and Table 3-2. This model is also used by Young and Bajorek [2] in their study of jet condensation.

Theofanous’ model was selected for use in the jet condensation model for its comprehensiveness and ease of implementation. The Theofanous model is developed for highly turbulent jets, so a separate model is needed for laminar and transition flow.

For laminar flow, a constant Nusselt number is used, similar to that used in laminar pipe flow. A constant Nusselt number of 4.364 is assumed, which is the same value used for laminar pipe flow. A linear ramp between the laminar heat transfer coefficient and the turbulent heat transfer coefficient is used in the transition region. This constant Nusselt number can also be modified to account for entrance effects in the jet similar to convective enhancement following a spacer grid in a reactor assembly. The entrance effects model is given in Equation 3.27
In the equation above, \( \frac{\text{Nu}}{\text{Nu}_0} \) represents the initial Nusselt number, and \( \frac{L}{D} \) is the local length of the jet divided by its diameter. Another laminar heat transfer model was investigated. A model developed by Isachenko [47] for liquid jet flow was also included in the liquid jet model. This expression is given in Equation 3.28

\[
St = a \left( \frac{L}{D} \right)^n \text{Re}^{-0.8} 
\]  

(3.28)

In Equation 3.28, \( a \) is a constant which depends on the both the length to diameter ratio of the jet and the length to diameter ratio of the nozzle of the jet, \( n \) is a constant which depends on the length to diameter ratio of the jet, \( \left( \frac{L}{D} \right) \) is the length to diameter ratio of the jet, and \( \text{Re} \) is the Reynolds number of the jet. The constant \( n \) is \(-0.88\) when the \( L/D \) is less than 12 or \(-0.57\) when the \( L/D \) is greater than 12. When the \( L/D \) of the jet is less than 12 and the \( L/D \) of the nozzle is near one, constant \( a \) is 0.24. For the same jet \( L/D \) but a larger nozzle length the constant becomes 0.185. When the \( L/D \) of the jet is greater than 12, the constant \( a \) can be either 0.125 or 0.088 for nozzle \( L/D \) of near one and for large ratios respectively.

The lower end of the transition region is assumed to be a Reynolds number slightly greater than the critical Reynolds number for pipe flow, which is generally accepted to be between 2300 and 2500. In this work, the high end of this is assumed, \( \text{Re} = 2500 \). The critical Reynolds number is assumed to be greater for jet flow because...
of the free surface. In a pipe, the walls act to contain turbulent energy. A free surface will allow the turbulent energy to dissipate more easily and therefore necessitate a higher Reynolds number flow to become turbulent. The upper range of the transition region is assumed to be at a Reynolds number of 4000. This is the same Reynolds number that is used as the lower limit for the turbulent eddy diffusivity by Travis [10].

The correlations that have been discussed for the liquid side heat transfer coefficient are used to determine the mass transfer rate, defined in Equation 3.29 as

$$\dot{\Gamma}^* = \frac{-\dot{q}_d - \dot{q}_w}{i_{fg}^*}$$  \hspace{1cm} (3.29)

When it is assumed that the vapor is not superheated this simplifies to Equation 3.30.

$$\dot{\Gamma}^* = \frac{-\dot{q}_d}{i_{fg}^*}$$  \hspace{1cm} (3.30)

In Equation 3.30 the liquid side heat transfer is given as Equation 3.31.

$$\dot{q}_d = \frac{h_{li} A_{HT}}{c_p} \left( i_f - i_{lj} \right)$$  \hspace{1cm} (3.31)

Correlations for the liquid side heat transfer coefficient have been previously discussed in this section. The enthalpies are taken at the new time values and the saturated liquid enthalpy is allowed to vary with pressure. The implicit mass transfer rate can be written as Equation 3.32

$$\dot{\Gamma} = \dot{\Gamma}^* \left( i_f - i_{lj} \right), \text{ where } \dot{\Gamma}^* = \frac{-h_{li} A_{HT}}{i_{fg}^* c_p}$$  \hspace{1cm} (3.32)
3.4.3 Sensible Heat Transfer

Sensible heat transfer will occur when there is a temperature gradient between the bulk vapor and the interface. Sensible heat flux is given in Equation 3-33 to be the product of the thermal conductivity of the vapor/gas mixture and the temperature gradient at the interface [20] as

$$ q''_i = k_v \left( \frac{dT}{dr} \right)_i $$  \hspace{1cm} (3-33)

Formulations for the liquid side heat transfer coefficient were given in the previous section. The heat transferred from the interface to the liquid is given in as

$$ \dot{q}_{il} = h_{il} A_{HT} (T_{sat} - T_{lj}) $$  \hspace{1cm} (3.34)

In the previous equation, $\dot{q}_{il}$ is the heat rate, $A_{HT}$ is the heat transfer area, $T_{sat}$ is the saturation temperature, and $T_{lj}$ is the bulk temperature of the liquid jet.

The heat transferred from the vapor to the interface is similarly given in Equation 3.35 as

$$ \dot{q}_{iv} = h_{iv} A_{HT} (T_{vb} - T_{sat}) $$  \hspace{1cm} (3.35)

In the above equation $\dot{q}_{iv}$ is the heat rate from the vapor to interface, $h_{iv}$ is the vapor side heat transfer coefficient, $A_{HT}$ is the heat transfer area, $T_{vb}$ is the vapor temperature, and $T_{sat}$ is again the saturation temperature. The vapor side heat transfer coefficient is taken
from the COBRA/TRAC theory manual for super heated vapor in a film heat transfer regime. The heat transfer coefficient is given as Equation 3.36

\[ h_v = \frac{f}{2} \rho c_p \mu_l |u| \Pr_v^{-\frac{3}{2}} \] 

The data set being used for validation does not include any experiments run with superheated vapor, so this has not been fully implemented into the stand alone model or the coupling with COBRA-TF. This also simplifies the coupling and allows the coupling to be explicit. It is however suggested for future work to include modeling superheated vapor to allow more flexibility in the implementation of the liquid jet model.

### 3.4.4 Turbulence within the Liquid Jet

According to Schlichting [9], for free turbulent flows, such as a free jet, the transverse gradients are large and the area of interest in the transverse direction is small compared to the main direction of flow. This means that the jet problem can be solved using boundary layer equations. Several boundary layer models were investigated for determining the temperature and velocity profiles within the liquid jet. Travis [10,11] developed a velocity and eddy diffusivity model for turbulent pipe flow. In his model, Travis developed a relationship for the ratio of the local fluid velocity to the maximum velocity in the flow. The relationship developed uses a zero velocity boundary condition at the wall and assumes the maximum velocity occurs at the center of the pipe. A temperature profile similar to the velocity profile will exist in the flow if the eddy
diffusivity of heat, $\varepsilon_H$, is close in value to the eddy diffusivity of momentum, $\varepsilon_M$. For this case to be valid, the Prandtl number must be close to one. The problem of fully turbulent pipe flow is somewhat similar in nature to the problem of a condensing jet being issued into a vapor environment. The condensate deposited at the interface has a very low velocity, which could create a situation similar to the no-slip condition at the wall of a pipe. The development of this type of condition would be dependent on having a sufficiently large mass deposition rate onto the jet.

Several researchers have investigated the turbulent nature of jet flows as turbulent free flow. Schlichting wrote that using Prandtl’s mixing length theory the solution of the turbulent circular jet takes the same form as a laminar circular jet with the kinematic viscosity replaced with the eddy diffusivity [9]. Using this relationship the axial velocity is given in Equation 3-37. In this relationship, $K$ is the constant kinematic momentum which is a measure of the strength of the jet.

$$u = \frac{3}{8\pi} \frac{K}{\varepsilon_0x} \frac{1}{\left(1 + \frac{1}{4}\eta^2\right)^2}$$

(3-37)

and

$$\eta = \frac{1}{4} \sqrt{\frac{3}{\pi}} \frac{\sqrt{K}}{\varepsilon_0} \frac{y}{x}$$

According to Schlichting, measurements by Reichardt provide a value of 0.0161 for the ratio of the eddy diffusivity to the constant kinematic momentum, $K$. 
Hinze [21, 22] discusses both the velocity and temperature distributions in a round free jet. The velocity profile in a jet is first solved for using classical theories and applying Prandtl’s mixing length theory. Transport of scalar quantities within the jet is also discussed. In Hinze’s model, the pressure outside of the jet region is assumed to be constant. Thus, the equation of motion of the jet, using Hinze’s nomenclature, becomes

\[ \rho \left( \overline{U}_x \frac{\partial \overline{U}_r}{\partial x} + \overline{U}_r \frac{\partial \overline{U}_x}{\partial r} \right) = \frac{1}{r} \frac{\partial}{\partial r} (r \sigma_{xr}) \]  \hspace{1cm} (3-38)

In Equation 3-38 \( \rho \) is the liquid density, \( \overline{U}_x \), is the average velocity in the axial direction, \( x \) is the coordinate in the axial direction, \( r \) is the coordinate in the radial direction, \( \sigma_{xr} \) is the interfacial shear.

Hinze solves this equation for the ratio of the velocity in the axial direction to the maximum velocity in the jet assuming that the turbulent transport of momentum can be characterized by the eddy diffusivity of momentum. The relationship for the velocity profile by Hinze is given as

\[ \frac{U}{U_m} = \left[ 1 + \frac{U_m r^2}{8 \varepsilon_m x} \right]^{-2} \]  \hspace{1cm} (3-39)

In Equation 3-39 \( U_m \) is the maximum velocity, \( U \) is the local velocity, \( \varepsilon_m \) is the eddy diffusivity of momentum, \( x \) and \( r \) are the coordinates in the axial and radial directions. This relationship assumes that the eddy diffusivity, \( \varepsilon_m \), is constant and the velocity of the fluid the jet is being injected into is very small compared to the jets own initial velocity. In the problem being investigated in this work, the vapor is quiescent when the jet is injected so this assumption is valid.
The equation of transport of a scalar quantity in a round free jet by Hinze is given as

\[
\bar{U}_x \frac{\partial \bar{\Gamma}}{\partial x} + \bar{U}_r \frac{\partial \bar{\Gamma}}{\partial r} = -\frac{1}{r} \frac{\partial}{\partial r} \left( ru, \gamma \right)
\]  \hfill (3-40)

This equation of transport is similar to the equation of motion of the jet given above, but here \( \Gamma \) is the scalar quantity being transported. In similar fashion to the case of momentum transport, the turbulent transport of the scalar quantity is assumed to be described using a thermal diffusivity coefficient, which is

\[
\overline{u, \gamma} = \epsilon_{\gamma} \frac{\partial \bar{\Gamma}}{\partial r}
\]  \hfill (3-41)

In Equation 3-41 \( \epsilon_{\gamma} \) is the eddy diffusivity of the scalar quantity being transported. In the case of the liquid jet this quantity would be temperature and \( \epsilon_{\gamma} \) would be \( \epsilon_H \), the eddy diffusivity of heat. Assuming that the ratio of the eddy diffusivity of momentum to the eddy diffusivity of heat is constant a relationship for the temperature profile within the jet in terms of the velocity profile within the jet can be determined to be

\[
\frac{\Gamma}{\Gamma_{\text{max}}} = \left( \frac{U}{U_{\text{max}}} \right)^{\epsilon_m/\epsilon_{\gamma}}
\]  \hfill (3-42)

It can be easily seen in Equation 3-42 that if the eddy diffusivities of heat and momentum are equal, the normalized velocity and temperature profiles within the jet will be equal.
Experiments by Corrsin and Uberoi [23] and Forthmann [24] confirm that Hinze’s method can be used to calculate the velocity profile and temperature profiles within air jets. But analysis of the Corrisin experiments by Hinze showed that for an air jet the eddy diffusivities of momentum and heat are not equal.

In his book, Hinze [22] suggests a model for the eddy diffusivity of momentum in a turbulent round jet of air which is given in Equation 3.43

\[ \varepsilon_m = 0.13u_0d_0 \]  \hspace{1cm} (3.43)

In the equation above \( u_0 \) and \( d_0 \) are the jet inlet velocity and diameter. Hinze’s model was developed from experiments in which an air jet is introduced into a quiescent air filled environment.

Both Travis [10] and Kutateladze [7] used data by Nikuradze [46] to develop models for the eddy diffusivity in jets of water. Isachenko, using Kutateladze, recommends a model for the eddy diffusivity of turbulent liquid jets [16]. This model is given in Equation 3.44

\[ \varepsilon_m = 0.00025u_0d_0 \]  \hspace{1cm} (3.44)

This correlation differs from the one developed by Hinze only by the coefficient, which is several orders of magnitude smaller. This most likely because a water jet issued into air or steam will not spread and mix as much as an air jet issued into air. According to Travis, Nikuradze’s data is valid down to a Reynolds number of 4000. Dividing Equation 3.44 by the kinematic viscosity results in a non-dimensional eddy diffusivity to
kinematic viscosity ratio in terms of the Reynolds number. This is given in Equation 3.45.

$$\frac{\varepsilon_m}{\nu} = 0.00025 \text{Re} \quad (3.45)$$

When the Reynolds number is 4000, the eddy diffusivity to kinematic viscosity ratio given in Equation 3.45 is equal to one.

### 3.4.5 Jet Formation

The liquid jet will be created through boundary conditions only. The boundary condition will include the jet velocity distribution, liquid temperature distribution, and radius of the nozzle through which the jet is introduced.

### 3.4.6 Jet Termination

The liquid jet can be destroyed by the jet impinging on a structure, or merging with the continuous liquid field. Jet impingement on a structure will not be treated in this work, as this does not occur in the region of interest for the experiments that are being studied. The liquid jet will only interface with the continuous liquid field in a computational cell which has a large liquid volume fraction, such as a jet impinging into a pool of liquid. The liquid jet will be merged with the continuous liquid field using mass, energy, and momentum sources.
3.4.7 Effects of Noncondensable Gas on Condensation

There have been a number of experiments and theoretical analysis done on the effects of noncondensable gases on condensation heat transfer rates. Much of this analysis has been done on condensation on vertical “cold” surfaces such as inside an externally cooled tube or on a cooled wall. The experiments \([25,26,27,28]\) for the most part focus on condensation within tubes. The experiments done at the University of Wisconsin [29] however were performed on a cooled wall to simulate a cooled containment wall. Several other experiments were performed on horizontal surfaces including on the outside of a cooled pipe [30], on the inside externally cooled pipes [31], and on downward facing surfaces [32] designed to simulate the containment wall.

The similarity between all of these experiments is that the condensate forms a film on a cold surface. Depending on the flow situation in the experiment, this film can either move in laminar or turbulent nature. As the vapor condenses on the liquid film a layer of noncondensable gas is left at the interface. The vapor must than diffuse through the gas layer to reach the liquid film as shown in Figure 3-1.
Many of the theoretical studies done in coordination with these experiments are focused on the diffusion of vapor through the gas layer and the buildup of the gas layer. Researchers have developed theoretical models to estimate the condensation heat transfer rate for both laminar [33] and turbulent [34] conditions. These and other diffusion layer theory models [20,35,36] rely on extending the Reynolds-Colburn analogy for heat and momentum transfer to mass and momentum transfer. For this to be accurate, both the equations and the boundary conditions must be identical [37]. In most cases the governing equations and most of the boundary condition are identical. The difficulty can
be in the boundary condition required for the transverse velocity at the film. In the mass transfer problem this velocity is usually not zero, while in the heat transfer case it is.

Hassan and Raja [38] performed simulations of two experiments using the RELAP5/MOD3 computer code. They concluded that the correlation for condensation heat transfer in the presence of noncondensable gases in that system code is inadequate. The RELAP5/MOD3 code uses a multiplication factor to reduce the condensation heat transfer rate when noncondensable gases are present. The reduction factor is given as

\[
x_{\text{cond}} = \frac{(p_v - p_{\text{min}})}{p^* f[\text{Re}_g]} F
\]

(3-46)

where

\[
f[\text{Re}_g] = \frac{5}{1 + 0.00001 \text{Re}_g}
\]

and

\[
F = \left[ 1 + f[\text{Re}_g] \exp \left[ -5 \left( \frac{p_g}{p} \right) \right] \right]
\]

In Equation 3-46 \( p_v \) is the partial pressure of steam, \( p_g \) is the partial pressure of air, \( p^* \) is the total pressure of the vapor/gas mixture, \( p_{\text{min}} \) is the minimum pressure in the steam tables, \( \text{Re}_g \) is the gas Reynolds number, which must be between zero and twenty thousand, and \( F \) is a function of the gas concentration. This model depends on the steam tables being used in the computer code and takes into account only the Reynolds
number of the gas, which is in contrast to other models found in literature that generally utilize the relative flow between the liquid and vapor.

Experiments and analysis have also been performed for condensation on liquid jets in the presence of noncondensable gases. These experiments have been discussed in Chapter 2. The correlation developed from these experiments and analysis is more relevant to the current work than that performed on films. Sklover and Rodivilin [4] developed a correlation for condensation suppression that is used as a basis for the model within the TRAC code [2], which is

\[
x_{\text{cond}} = 0.168 \left( \frac{\rho_L}{\rho_v} m \right)^{-0.1} \left( \frac{L}{D} \right)^{0.33}
\]  

(3-47)

In Equation 3-47, \( m \) is defined as the cooling ratio, which is the ratio of the mass flux of the liquid to the mass flux of steam. In the TRAC code and in Young and Bajorek’s work, assumptions are made that the vapor and liquid velocities are equal and the L/D ratio is one. The vapor velocity will be proportional to the condensation heat flux and not equal to the jet velocity. The actual L/D in Sklover and Rodivilin’s data varies between 15 and 60. These incorrect assumptions will lead to an incorrect calculation of the condensation suppression factor. Assuming a vapor velocity equal to the liquid velocity will most likely cause an overestimation of the condensation factor, but this is very much offset by the low value of L/D which will cause an underestimation of the suppression factor by a factor of 2.5 to four.
Young and Bajorek developed a model for condensation in the suppression of noncondensable gases that has the ability to adapt to both liquid film and liquid jet condensation environments. The condensation suppression factor is given as

\[ x_{\text{cond}}^2 - A(1 - x_{\text{cond}}) = 0 \] (3-48)

where

\[ A = \frac{\rho_v^2 i_{fg}}{h_{il}^2 (T_v - T_f) T_{sat}} \frac{C}{x_g} \frac{d_{vg}}{t^*} (1 - x_g) \frac{d_{vg}}{t^*} \] (3-49)

In Equation 3-48, \( x_{\text{cond}} \) is the condensation heat transfer suppression factor, \( x_g \) is the gas concentration, \( d_{vg} \) is the diffusion constant for vapor in the gas, \( t^* \) is the renewal time, \( \rho_v \) is the gas density, \( i_{fg} \) is the latent heat of vaporization, \( h_{il} \) is the liquid side heat transfer coefficient, \( T_v \) is the saturation temperature at the bulk vapor partial pressure, \( T_f \) is the liquid temperature, \( T_{sat} \) is the saturation temperature at the total pressure and constant \( C \) is given by

\[ \frac{P_v - P_{vi}}{P} = C \frac{T_v - T_{vi}}{T_{sat}} \] (3-49)

In Equation 3-49 \( P \) is the total pressure, \( T_{sat} \) is the saturation temperature at the total pressure, \( P_v \) is the bulk vapor partial pressure, \( P_{vi} \) is the vapor partial pressure at the interface, \( T_v \) is the saturation temperature at the bulk vapor partial pressure, and \( T_{vi} \) is the saturation temperature at the vapor partial pressure at the interface. The renewal time is given as a function of the hydraulic diameter and relative velocity of the system as

\[ t^* \approx 0.1 \frac{D_H}{U_r} \] (3.50)
3.4.8 Interfacial Shear

The interfacial shear force is calculated based on the shear model in COBRA-TF [44]. The interfacial force per unit volume is given in Equation 3.51.

\[ \tau_i'' = K_i u_{vl} \]  

(3.51)

In Equation 3.51 \( u_{vl} \) is the relative velocity between the vapor and the liquid. The coefficient \( K_i \) is dependent on the flow regime. The value of \( K_i \) for films is given in Equation 3.52.

\[ K_i = 2.0 \frac{f_i}{D_H} \rho_v |u_{vl}| \]  

(3.52)

In the equation for the interfacial friction coefficient, \( D_H \) is the hydraulic diameter, \( \rho_v \) is the vapor density, \( u_{vl} \) is again the relative velocity between the vapor and liquid and \( f_i \) is a coefficient of friction.

Based on a dimensional analysis of the interfacial shear forces, which follows, it is assumed that the effects of interfacial shear upon the liquid side of the interface can be neglected. The shear stress on both side of the interface must be equal as

\[ \tau_v = \tau_f \]  

(3.53)

From the definition of the shear stress it follows that Equation 3.54

\[ \mu_v \frac{\partial u}{\partial r} |_v = \mu_v \frac{\partial u}{\partial r} |_l \]  

(3.54)
Dimensional analysis allows that the change in the velocities are equal to the change from the bulk velocity to the interface velocity and the differential in the radial direction is equal to the boundary layer thickness. Equation 3.54 becomes

\[ \frac{\mu_v}{\delta_v} (u_i - u_v) = \frac{\mu_f}{\delta_f} (u_f - u_i) \]  

Equation 3.55

Re-arranging terms in Equation 3.55 gives

\[ \frac{u_f - u_i}{u_i - u_v} = \frac{\mu_v \delta_f}{\mu_f \delta_v} \]  

Equation 3.56

In Equation 3.56, the ratio of the vapor viscosity to the liquid viscosity is very small as should be the ratio of the liquid boundary layer to the vapor boundary layer. It then follows that

\[ \frac{u_f - u_i}{u_i - u_v} \ll 1 \]  

Equation 3.57

Therefore Equation 3.58

\[ u_f = u_i \]  

Equation 3.58

3.5 Summary of Selected Constitutive Models

The constitutive relationships needed to solve the set of liquid jet equations were described above. For many of the relationships, several choices of empirical and/or theoretical models were discussed. This section will summarize which models where selected for use in the liquid jet model. The chosen constitutive models are summarized in Table 3-3 below.
Table 3-3: Summary of constitutive models

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Model</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turbulent Heat Transfer</td>
<td>$h_{ij} = 0.02 \rho c_p u \left( \frac{x}{D} \right)^{-0.5}$</td>
<td>Theofanous [45]</td>
</tr>
<tr>
<td>Laminar Heat Transfer</td>
<td>$St = a \left( \frac{L}{D} \right)^n Re^{-0.8}$</td>
<td>Isachenko [47]</td>
</tr>
<tr>
<td>Turbulent Eddy Diffusivity</td>
<td>$\epsilon_M = 0.00025 u_g d_0$</td>
<td>Isachenko [16]</td>
</tr>
<tr>
<td>Condensation Suppression in the Presence of Noncondensable gas</td>
<td>$x_{cond} = 0.168 \left( \frac{\rho_L m}{\rho_v} \right)^{-0.1} \left( \frac{L}{D} \right)^{0.33}$</td>
<td>Sklover [4]</td>
</tr>
<tr>
<td>Interfacial Shear Force</td>
<td>$K_I = 2.0 \frac{f_I}{D_H} \rho_v</td>
<td>u_{ij}</td>
</tr>
</tbody>
</table>

The Theofanous model for turbulent heat transfer was selected for use in the liquid jet model because it was suggested by Theofanous after reviewing all of the then available jet condensation heat transfer correlations. The laminar heat transfer coefficient by Isachenko was the only heat transfer coefficient that was developed specifically for condensation on water jets. The eddy diffusivity model by Isachenko was chosen because it provided the most accurate comparison to the experimental data compared to those by Hinze and Travis. The correlation for condensation suppression in the presence of noncondensable gas by Sklover was selected over that by Young and Bajorek because of its ease of implementation. Both models have been included in the code, but for the reason previously stated the model by Sklover is used by default. The interfacial shear model from COBRA-TF was selected for its simplicity in implementing.
Chapter 4

Linearization and Implementation of Equations of Motion

The flow equations introduced in Chapter 3 are presented again here in greater detail. In this chapter the method used to go from the differential equations to a set of equations that can be implemented and solved numerically will be explored.

Both the liquid jet and vapor boundary layer flow equations have similar forms and are linearized in the same way. Therefore this section will describe the linearization of the equations in a general way that is applicable to both fluid regions. In the following equations a superscript \( n \) designates an old time value. A superscript asterisk designates a donor cell old time value in the semi-implicit formulation. The linearization of the axial momentum equation will be presented first.

A two dimensional representation of a computational cell is picture below in Figure 4-1. In the following discussion of the equation linearization several terms are used to describe the locations at which quantities are used. The cell centered values are the average values within the computational cell and for computational purposes are located at point \((i,j)\) in Figure 4-1. Face values exist at locations O, I, T, and B in the Figure below. The axial velocities are solved at the top, T, and bottom, B, faces using a staggered mesh which is shown in Figure 4-2 overlayed the continuity cells. The axial momentum computational cells are shown shaded. The radial velocities are solved at the
inside, I, and outside, O, faces again using a staggered mesh. This is shown overlaying the continuity cells in Figure 4-3. Donor cell values, depending on the direction of the old time velocity, are taken from either the cell center of the neighboring cell or the cell centered value of the cell in which the equation is being solved. For example, if the axial velocity at the bottom face, shown is B in the figure below, is positive then the donor cell values in that term are taken from cell (i,j). If the axial velocity at the same face were negative the values would be taken from cell (i+1,j).

Figure 4-1: Two-Dimensional depiction of a computational cell and its neighbors
Figure 4-2: Axial momentum staggered mesh

Figure 4-3: Radial momentum staggered mesh
4.1 Axial Momentum Equation Linearization

The axial momentum equation is solved semi-implicitly, with the convecting velocities explicit and the time derivative, pressure force, and interfacial shear terms using new time values. The two dimensional form of the axial momentum equation is given as Eq. 4.1

\[
\frac{\partial \rho u}{\partial t} + \frac{\partial \rho uu}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} \left( \rho u vr \right) = \dot{\Gamma} u + \rho f_i + \rho g - \frac{\partial p}{\partial x} + \rho \varepsilon_m \nabla^2 u
\]  

(4.1)

In the above equation, the product rule is applied to the time rate change of momentum term in Eq. 4.2

\[
\frac{\partial \rho u}{\partial t} = \rho \frac{\partial u}{\partial t} + u \frac{\partial \rho}{\partial t}
\]  

(4.2)

The density is both a function of pressure and enthalpy and can be expanded as in Eq. 4.3

\[
\frac{\partial \rho}{\partial t} = \frac{\partial \rho}{\partial i} \frac{\partial i}{\partial t} + \frac{\partial \rho}{\partial p} \frac{\partial p}{\partial t}
\]  

(4.3)

The derivatives of the density with respect to enthalpy and pressure are found from the equation of state and are taken at the old time.

The derivative of the enthalpy with respect to time is expanded in Eq. 4.4
And the derivative of the pressure with respect to time is expanded in Eq. 4.5

\[
\frac{\partial p}{\partial t} = \frac{p_{i,j} - p_{i,j}^n}{\Delta t}
\] (4.5)

The derivative of the axial momentum in the axial direction is given in Eq. 4.6

\[
\frac{\partial (\rho u)^*}{\partial x} = \frac{(\rho u)_B^* u_B^* - (\rho u)_T^* u_T^*}{\Delta x}
\] (4.6)

In the above term, the momentum being fluxed is a donor cell momentum and the velocity at the face is found by using a linear average of the velocities on either side. The subscript B denotes the bottom edge of the momentum cell and the subscript T represents the top edge of the momentum cell.

The radial derivative of the axial momentum is expanded using the chain rule in Eq. 4.7

\[
\frac{1}{r} \frac{\partial}{\partial r} [\rho u v r] = \rho u \frac{\partial v}{\partial r} + \rho v \frac{\partial u}{\partial r} + \rho v \frac{\partial \rho}{\partial r} + \frac{\rho uv}{r}
\] (4.7)

This term can then be written as Eq. 4.8

\[
\frac{1}{r} \frac{\partial}{\partial r} [\rho u v r]
= \rho_{i,j} u_{i,j}^n \frac{v_{i,j}^o - v_{i,j}^n}{\Delta r} + \rho_{i,j} v_{i,j}^n \frac{u_{i,j}^o - u_{i,j}^n}{\Delta r} + \rho_{i,j} v_{i,j}^n \frac{\rho_o - \rho_t}{\Delta r} + \rho_{i,j} u_{i,j}^n v_{i,j}^n \frac{\rho_{i,j}}{r_{i,j}}
\] (4.8)
Again in the radial derivative, as in the axial derivative, the values being fluxed are donor cell values. The radial velocity at the cell edge is a linear average of the two radial velocities that share part of the face. The cell centered radial velocity in the second term of the radial derivative is found by linear average, as are the density and radius.

The axial momentum source from mass transfer is defined as the product of the mass transfer rate and a mass transfer velocity as shown in Eq. 4.9. This term is only present at the interface between the vapor boundary layer and the liquid jet. When condensation is occurring the mass transfer rate will be negative. The velocity associated with the mass transfer will also depend on its location. In the vapor, when condensing, the velocity will be the axial velocity within the cell. In the liquid jet, when condensing, the velocity will be the axial velocity in the vapor.

\[ \Gamma^{\omega} u = \Gamma^{\omega}_{i,j} u^\Gamma_{i,j} \]  

(4.9)

The mass transfer rate in the axial momentum cell will be the linear average of the mass transfer rates in the continuity cells surrounding it. It is assumed that the momentum addition to the liquid jet from mass transfer is small compared to the momentum of the jet and is ignored.

The interfacial shear term uses a cell centered density found by linear average. The shear force is calculated in the method discussed in 3.4.8. The velocities in the interfacial shear term are taken at the new time value. This is done to aid in the stability...
of the solution scheme. The interfacial shear term, as applied at the interface in the vapor boundary layer is given in Eq. 4.10

\[ u_{ib} = u_{ib} + K_I (u_{ij} - u_{ib}) \]  \hspace{1cm} (4.10)

In Eq. 4.10 all of the velocity terms are new time value velocities. By neglecting the effects of the interfacial shear forces on the liquid side of the interface, the solution matrix is simplified.

The body force due to gravity uses a cell centered density found using a linear average.

The pressure force term is shown in Eq. 4.11.

\[ \frac{\partial p}{\partial x} = \frac{p_{t,j} - p_{t-1,j}}{\Delta x} \]  \hspace{1cm} (4.11)

In this equation the pressures are new time values at the edges of the axial momentum cell. This means that they are the cell centered values in the continuity cells at either side of the momentum cell.

The final term in the axial momentum equation is the diffusion term, which can be expanded into Eq. 4.12.

\[ \rho \varepsilon_M \nabla^2 u = \varepsilon_M \left[ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} \right] \]  \hspace{1cm} (4.12)
The diffusion of axial momentum can be either molecular or turbulent. This is determined by summing the turbulent momentum diffusivity calculated with the viscosity of the fluid. This is used as the momentum diffusivity, $\varepsilon_M$. The diffusion term is given in Eq. 4.13

$$
\rho \varepsilon_M \left[ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} \right]
= \varepsilon_M \left[ \frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{\Delta x^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{\Delta r^2} + \frac{u_{i,j+1}^n - u_{i,j}^n}{2r_{i,j} \Delta r} \right]
$$

(4.13)

The densities used in the diffusion term are cell centered and are found using linear averages. The velocities used are also cell centered.

The full axial momentum equation for a non-interface is written in Eq. 4.14

$$
\rho_{i,j} \frac{u_{i,j}^n - u_{i,j}^n}{\Delta t} + u_{i,j}^n \left[ \frac{\partial}{\partial t} \left( \rho u \right)_{i,j}^n + \frac{\partial}{\partial r} \left( \rho v \right)_{i,j}^n + \frac{\partial}{\partial r} \left( \rho \right)_{i,j}^n \frac{p_{i,j}^n - p_{i,j}^n}{\Delta r} \right]
+ \rho_{i,j} \left( \rho u \right)_{i,j}^n \frac{u_{i,j}^n - u_{i,j}^n}{\Delta r} + \rho_{i,j} \left( \rho v \right)_{i,j}^n \frac{v_{i,j}^n - v_{i,j}^n}{\Delta r} + \rho_{i,j} \frac{p_{i,j}^n - p_{i,j}^n}{r_{i,j}}
= \rho_{i,j} g - \frac{p_{i,j} - p_{i,j}}{\Delta x}
+ \rho_{i,j} \varepsilon_M \left[ \frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{\Delta x^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{\Delta r^2} + \frac{u_{i,j+1}^n - u_{i,j}^n}{2r_{i,j} \Delta r} \right]
$$

(4.14)
The axial momentum equation is solved for the new time velocity in Eq. 4.15

\[
\begin{align*}
    u_{i,j} &= u_{i,j}^n - u_{i,j}^n \left[ \left( \frac{\partial \rho}{\partial i} \right)_{i,j} + \left( \frac{\partial \rho}{\partial p} \right)_{i,j} \frac{p_{i,j} - p_{i,j}^n}{\rho_{i,j}} \right] \\
    &\quad - \Delta t \left( \rho u \right)_{i,j}^n \frac{u_{i,j}^n - u_{i,j}^n}{\rho_{i,j}^n} - u_{i,j}^n \Delta t \frac{v_{o,i,j}^n - v_{i,j}^n}{\Delta r} \\
    &\quad - v_{i,j}^n \Delta t \frac{u_{o,i,j}^n - u_{i,j}^n}{\Delta r} - u_{i,j}^n \Delta t \frac{\rho_{o,i,j}^n - \rho_{i,j}^n}{\rho_{i,j}^n} - \Delta t \frac{u_{i,j}^n v_{o,i,j}^n}{r_{i,j}^n} \\
    &\quad + g \Delta t - \frac{p_{i,j}^n - p_{i-1,j}^n}{\rho_{i,j}^n} \Delta t \\
    &\quad + \varepsilon_M \Delta t \left[ \frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{\Delta x^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{\Delta r^2} \\
    &\quad + \frac{u_{i,j+1}^n - u_{i,j-1}^n}{2r_{i,j} \Delta r} \right]
\end{align*}
\]

(4.15)

The axial momentum equation is broken up into explicit and implicit terms in Eq. 4.16 and Eq. 4.17 below.

\[
\begin{align*}
    u_{i,j} &= u_{i,j}^n + c_{i,j}^0 \left[ p_{i,j} - p_{i-1,j} \right] + c_{i,j}^1 \left[ p_{i,j} + p_{i-1,j} \right] + c_{i,j}^2 \left[ i_{i,j} + i_{i-1,j} \right] \\
\end{align*}
\]

(4.16)

where
In Eq. 4.16, the coefficients are cell centered on the momentum cell and the pressures and enthalpies are cell edge values which are cell centered values in the corresponding continuity cells. In the third and fourth terms on the right hand side, those with $c_{i,j}^1$ and $c_{i,j}^2$, the pressure and enthalpy at the center of the momentum cell are found using a linear average of the values at the center of the surrounding continuity cells.

At the vapor side of the vapor/liquid interface, the axial momentum equation includes implicit interfacial shear terms. For this location, the full axial momentum equation is written in Eq. 4.18

\[
\begin{align*}
  u_{i,j}^0 &= u_{i,j}^n + u_{i,j}^n \left[ \left( \frac{\partial \rho}{\partial i} \right)_{i,j} \rho_{i,j} + \left( \frac{\partial \rho}{\partial j} \right)_{i,j} \rho_{i,j} \right] \\
  -\Delta t \left( \frac{\rho u}{\rho} \right)_B \left( u^n + \frac{\partial u}{\partial t} \right)_{i,j} - u_{i,j} \Delta t \frac{v^n_{i,j} - v^n_j}{\Delta r} \\
  -v_{i,j} \Delta t \frac{u^n_{i,j} - u^n_j}{\Delta r} - u_{i,j}^n \Delta t \frac{\rho_o - \rho_l}{\Delta r} - \Delta t \frac{u_{i,j}^n v_{i,j}^n}{r_{i,j}} \\
  + g \Delta t \\
  + \epsilon_M \Delta t \\
  \left[ \frac{u_{i,j}^n - 2u_{i,j} + u_{i-1,j}^n}{\Delta x^2} + \frac{u_{i,j}^n + u_{i,j+1}^n - 2u_{i,j}^n}{\Delta x^2 \Delta r} \\
  + \frac{u_{i,j+1}^n - u_{i,j-1}^n}{2r_{i,j} \Delta r} \right]
\end{align*}
\]  

(4.17)
The new time axial velocity can be written in terms of the new time axial velocities on both sides of the interface as Eq. 4.19

\[
\rho_{i,j} \frac{u_{i,j} - u_{i,j}^n}{\Delta t} + u_{i,j}^n \left[ \left( \frac{\partial \rho}{\partial t} \right)_{i,j}^n + \left( \frac{\partial \rho}{\partial p} \right)_{i,j}^n \left( p_{i,j} - p_{i,j}^n \right) \right]
\]

\[
+ \left( \rho u_b^n \right) \frac{u_b^n - \left( \rho u_j^n \right)}{\Delta x} + \rho_{i,j} u_{i,j}^n \frac{v_0^n - v_j^n}{\Delta r}
\]

\[
+ \rho_{i,j} v_{i,j} \frac{u_{i,j}^n - u_{i,j}^n}{\Delta r} + u_{i,j}^n v_{i,j} \left( \rho_0 - \rho_l \right) + p_{i,j} u_{i,j}^n \frac{v_{i,j}^n}{r_{i,j}}
\]

\[
= \rho_{i,j} K_l \left( u_{i,j+1} - u_{i,j} \right) + \rho_{i,j} \delta \frac{p_{i,j} - p_{i,j-1}}{\Delta x}
\]

\[
+ \rho_{i,j} \varepsilon_M \left[ \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{\Delta x^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{\Delta r^2}
\]

\[
+ \frac{u_{i,j+1}^n - u_{i,j-1}^n}{2r_{i,j} \Delta r}
\]

The new time axial velocity can be written in terms of the new time axial velocities on both sides of the interface as Eq. 4.19

\[
u_{i,j} = u_{i,j} \left[ 1 - K_j \right] + u_{i,j+1} K_j
\]

In Eq. 4.19, the velocity on the left hand side is the new time velocity in the cell with interfacial shear accounted for. The velocities on the right hand side are the velocity at the interface in the vapor, \( u_{i,j} \), and the liquid jet, \( u_{i,j+1} \), which are found using Eq. 4.15. The resulting formulation, when broken into implicit and explicit terms is given in Eq. 4.20.
In the solution sequence used, the coefficients given in Eq. 4.17 are found using old time values. The formulation in Eq. 4.16 or Eq. 4.20, depending on location, is inserted for the new time value axial velocity in the continuity, energy, and non-condensable gas transport equations. The new time velocity is then found after the solution of those equations using Eq. 4.16 or Eq. 4.20 at the vapor side of the interface. The radial momentum equation, which will be discussed next, is solved using a similar methodology, but without the interfacial shear terms.

4.2 Radial Momentum Equation Linearization

The radial momentum equation, like the axial momentum equation, is solved semi-implicitly with only the time derivative and pressure terms using new time values. The two dimensional form of the radial momentum equation in cylindrical coordinates is given in Eq. 4.21

\[
\frac{\partial \rho v}{\partial t} + \frac{\partial \rho vu}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} \left[ \rho vvr \right] = \Gamma v - \frac{\partial p}{\partial r} + \rho \varepsilon_M \left[ \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial r^2} + \frac{1}{r} \frac{\partial v}{\partial r} \right]
\] (4.21)
The time derivative term in the radial momentum equation can be expanded in the same fashion as the axial momentum equation. The time derivative is expanded in Eq. 4.22

$$\frac{\partial \rho v}{\partial t} = \rho \frac{\partial v}{\partial t} + v \frac{\partial \rho}{\partial t} \quad (4.22)$$

The density is both pressure and enthalpy dependant and as such can be expanded using chain rule. The resulting expanded time derivative is given in Eq. 4.23

$$\frac{\partial \rho v}{\partial t} = \rho \frac{\partial v}{\partial t} + v \left[ \frac{\partial \rho}{\partial i} \frac{\partial i}{\partial t} + \frac{\partial \rho}{\partial p} \frac{\partial p}{\partial t} \right] \quad (4.23)$$

The time derivative can finally be expanded in Eq. 4.24

$$\frac{\partial \rho v}{\partial t} = \rho_{i,j} \frac{v_{i,j} - v_{i,j}^{n}}{\Delta t} + v_{i,j}^{n} \left[ \left( \frac{\partial \rho}{\partial i} \right)_{i,j}^{n} i_{i,j} - i_{i,j}^{n} + \left( \frac{\partial \rho}{\partial p} \right)_{i,j}^{n} p_{i,j} - p_{i,j}^{n} \right] \quad (4.24)$$

The axial derivative of the radial momentum is written in Eq. 4.25

$$\frac{\partial \rho vu}{\partial x} = \frac{(\rho v)^{n}_{i} u_{i}^{n} - (\rho v)^{n}_{i+1/2} u_{i+1/2}^{n}}{\Delta x} \quad (4.25)$$

In the above term the subscript $B$ refers to the bottom, or $i + \frac{1}{2}$, face of the axial momentum cell while the subscript $T$ refers to the top, $i - \frac{1}{2}$, face. In this term of the radial momentum equation the radial momentum, $(\rho v)$, which is a donor cell momentum, is fluxed by the axial velocity at the edges of the radial momentum cell. The axial velocity at those faces is found as a linear average of the two velocities that’s share the edge.
The radial derivative of the radial momentum is given in Eq. 4.26

\[
\frac{1}{r} \frac{\partial}{\partial r} \left[ \rho v v r \right] = \rho v v \frac{\partial v}{\partial r} + \rho v v \frac{\partial v}{\partial r} + v v \frac{\partial v}{\partial r} + \frac{\rho v v}{r} \quad (4.26)
\]

This expansion is performed using the product rule and is expanded in finite volume nomenclature in Eq. 4.27.

\[
\frac{1}{r} \frac{\partial}{\partial r} \left[ \rho v v r \right] = 2 \rho_{i,j} v_{i,j}^v \frac{v_{i,j}^v - v_{i}^v}{\Delta r} + v_{i,j}^v v_{i,j}^v \frac{\rho_{i}^v - \rho_{j}^v}{\Delta r} + \frac{\rho_{i,j} v_{i,j} v_{i,j}}{r_{i,j}} \quad (4.27)
\]

In the above equation the subscript \( O \) refers to the outside, \( j + \frac{1}{2} \), face of the radial momentum cell and the subscript \( I \) refers to the inside, \( j - \frac{1}{2} \), face. The face value is found by linear average of the cell centered value on either side of the face. The cell centered density in the radial derivative term is found by linear average of the densities in the continuity cells on either side of the radial momentum cell.

The radial momentum source due to mass transfer is given in Eq. 4.28

\[
\dot{\Gamma}^\omega = \dot{\Gamma}_{i,j}^{\omega, r_{i,j}} \quad (4.28)
\]

The mass transfer rate in the radial momentum cell is found by linear average. The mass transfer radial velocity is a function of its location. In the vapor boundary layer, in the case of condensation, the mass transfer radial velocity is the radial velocity of the vapor. In the liquid jet, in the case of condensation, the mass transfer radial momentum is neglected.
The radial pressure force term is written in Eq. 4.29

\[
\frac{\partial p_i}{\partial r} = \frac{p_{i,j} - p_{i,j-1}}{\Delta r}
\]  

(4.29)

In the radial pressure force term the pressures are given at the edges of the radial momentum cell, or at the center of the continuity cells.

The diffusion of radial momentum is given in Eq. 4.30

\[
\rho \varepsilon_M \left[ \frac{\partial^2 v_i}{\partial x^2} + \frac{\partial^2 v_i}{\partial r^2} + \frac{1}{r} \frac{\partial v_i}{\partial r} \right] = \rho_{i,j} \varepsilon_{M_{i,j}} \left[ \frac{v_{i+1,j} - 2v_{i,j} + v_{i-1,j}}{\Delta x^2} + \frac{v_{i,j+1} - 2v_{i,j} + v_{i,j-1}}{\Delta r^2} + \frac{v_{i,j+1} - v_{i,j-1}}{2r_{i,j} \Delta r} \right]
\]  

(4.30)

In the above term the density and momentum diffusivity are cell centered properties. The density is found by linear average. The momentum diffusivity is given as the sum of the fluid viscosity and the turbulent diffusivity. The radial velocities are also cell centered quantities.

The entire linearized radial momentum equation is given in Eq. 4.31

\[
\frac{\rho \varepsilon_M}{\Delta t} v_{i,j} - v_{i,j}^{\text{in}} \left[ \frac{\partial p_i}{\partial t} \right]_{i,j} + \frac{\rho v_{i,j}^{\text{in}}}{\Delta x} \left( \frac{\partial v_i}{\partial t} \right)_{i,j} + \frac{(\rho v)^{\text{T}} u_i^{\text{T}} - (\rho v)^{\text{B}} u_i^{\text{B}}}{\Delta x} + 2\rho v_{i,j}^{\text{in}} \left( \frac{\partial v_i}{\partial r} \right)_{i,j} + \rho v_{i,j}^{\text{in}} \left( \frac{\partial v_i}{\partial r} \right)_{i,j} = \frac{\Delta v_{i,j}}{\Delta r} - \frac{p_{i,j} - p_{i,j-1}}{\Delta r}
\]  

(4.31)
The radial momentum equation is solved for the new time radial velocity in Eq. 4.32

\[ v_{i,j} = v_{i,j}^0 - v_{i,j}^0 \left[ \left( \frac{\partial \rho}{\partial t} \right)_{i,j}^n + \left( \frac{\partial \rho}{\partial \rho} \right)_{i,j}^n \right] \]

\[-\Delta t \left( \frac{\rho v_x}{\rho} \right)_{i,j}^n u_b^x - \left( \rho v_x^0 \right)_{i,j}^n u_r^x - 2v_{i,j}^0 \Delta t \frac{v_{i,j}^0 - v_{i,j}^0}{\Delta r} - v_{i,j}^0 v_{i,j}^0 \Delta t \frac{\rho_{i,j}^0 - \rho_{i,j}^0}{\rho_{i,j}^0 \Delta r} \]

\[-\frac{v_{i,j}^0 v_{i,j}^0}{r_{i,j}} - \frac{1}{r_{i,j}} + \frac{\nabla^0}{\rho_{i,j}} \Delta t \]

\[ + \frac{\nabla^0}{\rho_{i,j}} \Delta t \left[ \frac{v_{i,j}^0 v_{i,j}^0 - 2v_{i,j}^0 + v_{i,j}^0}{\Delta x^2} + \frac{v_{i,j}^0 v_{i,j}^0 - 2v_{i,j}^0 + v_{i,j}^0}{\Delta r^2} + \frac{v_{i,j}^0 v_{i,j}^0 - 2v_{i,j}^0 + v_{i,j}^0}{2r_{i,j} \Delta r} \right] \]

(4.32)

The radial momentum equation, solved for the new time velocity, is broken up into its explicit and implicit components in Eq. 4.33 and Eq. 4.34 below.

\[ v_{i,j} = v_{i,j}^0 + d_{i,j}^0 \left[ p_{i,j} - p_{i,j-1} \right] + d_{i,j}^1 \left[ p_{i,j} + p_{i,j-1} \right] + d_{i,j}^2 \left[ i_{i,j} + i_{i,j-1} \right] \]

(4.33)

where

\[ v_{i,j}^0 = v_{i,j}^0 + v_{i,j}^0 \left[ \left( \frac{\partial \rho}{\partial t} \right)_{i,j}^n + \left( \frac{\partial \rho}{\partial \rho} \right)_{i,j}^n \right] \]

\[-\Delta t \left( \frac{\rho v_x}{\rho} \right)_{i,j}^n u_b^x - \left( \rho v_x^0 \right)_{i,j}^n u_r^x - 2v_{i,j}^0 \Delta t \frac{v_{i,j}^0 - v_{i,j}^0}{\Delta r} - v_{i,j}^0 v_{i,j}^0 \Delta t \frac{\rho_{i,j}^0 - \rho_{i,j}^0}{\rho_{i,j}^0 \Delta r} \]

\[-\frac{v_{i,j}^0 v_{i,j}^0}{r_{i,j}} - \frac{1}{r_{i,j}} + \frac{\nabla^0}{\rho_{i,j}} \Delta t \]

\[ + \frac{\nabla^0}{\rho_{i,j}} \Delta t \left[ \frac{v_{i,j}^0 v_{i,j}^0 - 2v_{i,j}^0 + v_{i,j}^0}{\Delta x^2} + \frac{v_{i,j}^0 v_{i,j}^0 - 2v_{i,j}^0 + v_{i,j}^0}{\Delta r^2} + \frac{v_{i,j}^0 v_{i,j}^0 - 2v_{i,j}^0 + v_{i,j}^0}{2r_{i,j} \Delta r} \right] \]

(4.34)
In Eq. 4.33, the coefficients are cell centered on the momentum cell and the pressures and enthalpies are cell edge values which are cell centered values in the corresponding continuity cells. In the third and fourth terms on the right hand side, those with \( d_{i,j}^1 \) and \( d_{i,j}^2 \), the pressure and enthalpy at the center of the momentum cell are found using a linear average of the values at the center of the surrounding continuity cells.

In the solution sequence used, the coefficients given in Eq. 4.34 are found using old time values. The formulation in Eq. 4.33 is inserted for the new time value radial velocity in the continuity, energy, and non-condensable gas transport equations. The new time velocity is then found after the solution of those equations using Eq. 4.33.

### 4.3 Continuity Equation Linearization

The continuity equation, like the momentum equations is solved in a semi-implicit method. In this solution scheme, the only terms that are new time values, other than in the time derivative, are the velocities in the convection terms and the mass transfer rate. The two-dimensional continuity equation in cylindrical coordinates is given in Eq. 4.35

\[
\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} \left( \rho vr \right) = \Gamma^\sigma
\]  

(4.35)

The density is a function of both the pressure and enthalpy. The time derivative of the density is re-written so show this in Eq. 4.36
\[
\frac{\partial \rho}{\partial t} = \frac{\partial \rho}{\partial t} \frac{\partial p}{\partial t} + \frac{\partial \rho}{\partial t} \frac{\partial p}{\partial t}
\]  \hspace{1cm} (4.36)

In the term above, the derivatives of the density with respect to the enthalpy and pressure are found from the equation of state. The finite volume formulation of the time derivative term is given in Eq. 4.37

\[
\frac{\partial \rho}{\partial t} = \left( \frac{\partial \rho}{\partial t} \right)_{i,j} \frac{i_{i,j} - i_{i-1,j}}{\Delta t} + \left( \frac{\partial \rho}{\partial t} \right)_{i,j} \frac{p_{i,j} - p_{i-1,j}}{\Delta t}
\]  \hspace{1cm} (4.37)

All of the values in the time derivative term written above are cell centered values given in the continuity cell.

The axial derivative of the continuity equation is expanded in Eq. 4.38

\[
\frac{\partial \rho}{\partial x} = \frac{\rho_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} - \rho_{i-\frac{1}{2},j} u_{i-\frac{1}{2},j}}{\Delta x}
\]  \hspace{1cm} (4.38)

In the axial derivative, the density, a cell centered donor cell quantity, is fluxed by the axial velocity at the axial faces of the continuity cell. The subscript \( B \) refers to the bottom, or \( i + \frac{1}{2} \), face of the continuity cell, while the subscript \( T \) refers to the top, or \( i - \frac{1}{2} \), face of the cell. The axial velocities at the faces are the cell centered axial velocities in the axial momentum cell mesh.

The radial derivative of the continuity equation, after expansion using product rule, is given in Eq. 4.39
The radial derivative given in Eq. \(4.39\) is further expanded, into its finite volume formulation, in Eq. \(4.40\)

\[
\frac{1}{r} \frac{\partial}{\partial r} \left[ \rho v_r \right] = \rho \frac{\partial v}{\partial r} + v \frac{\partial \rho}{\partial r} + \frac{\rho v}{r} \tag{4.39}
\]

The mass source from mass transfer is a cell centered value in the continuity cell that is found in the method described in \(3.4.2\). The mass transfer is treated implicitly such that the new time values are used for the vapor and liquid enthalpies. The general form of the mass transfer rate is given in Eq. \(4.41\) below. In this equation \(\Gamma^*\) is found using old time values and the liquid and interface enthalpies are new time values.

\[
\Gamma = \Gamma ^* \left( i_j - i_{i,j} \right), \quad \text{where} \quad \Gamma ^* = - \frac{h_{ij} A_{ij}}{i_{jk} c_p} \tag{4.41}
\]
In Eq. 4.41 above, \( h_{li} \) is the liquid side heat transfer coefficient, \( A_{HT} \) is the heat transfer area, \( \dot{\epsilon}_{fg} \) is the latent heat of vaporization, which in the case of condensation with saturated vapor is given as the difference between the vapor enthalpy and the saturated liquid enthalpy, and \( c_p \) is the specific heat at constant pressure. The formulation for \( \Gamma^* \) given in Eq. 4.41 assumes that there is no superheat and therefore no vapor side heat transfer.

The finite volume formulation of the continuity equation is given in Eq. 4.42

\[
\left( \frac{\partial \rho}{\partial t} \right)_{i,j} + \frac{\rho_{i,j} - \rho_{i-1,j}}{\Delta t} + \frac{\partial \rho}{\partial p}_{i,j} + \frac{p_{i,j} - p_{i-1,j}}{\Delta t} + \frac{\rho_{t,i,j} - \rho_{t,i-1,j}}{\Delta x} + \frac{v_{i,j} - v_{i-1,j}}{\Delta r} + v_{i,j} \left[ \frac{\rho_{o,i,j} - \rho_{l,i,j}}{\Delta r} \right] \right) = \dot{\Gamma}_{i,j}^\rho
\]

(4.42)

At the vapor side of the vapor/jet interface, Eq. 4.42 becomes Eq. 4.43 given below.

\[
\left( \frac{\partial \rho}{\partial t} \right)_{i,j} + \frac{\rho_{i,j} - \rho_{i-1,j}}{\Delta t} + \frac{\partial \rho}{\partial p}_{i,j} + \frac{p_{i,j} - p_{i-1,j}}{\Delta t} + \frac{\rho_{t,i,j} - \rho_{t,i-1,j}}{\Delta x} + \frac{v_{i,j} - v_{i-1,j}}{\Delta r} + v_{i,j} \left[ \frac{\rho_{o,i,j} - \rho_{l,i,j}}{\Delta r} \right] \right) = \dot{\Gamma}_{i,j}^\rho \left( i_{f,i,j} - i_{i,j-1} \right)
\]

(4.43)

At the liquid side of the interface, Eq. 4.42 becomes Eq. 4.44 below.
A negative mass transfer rate indicates condensation, so on the liquid side of the interface, as in Eq. 4.44 the term is negative when on the right hand side of the equation. In the two previous equations the liquid saturation is taken to be at the new time value. However at the time of calculation, the fluid properties are only known at the old time value. The derivative of the saturation enthalpy with pressure is also known and can be used to find the new time value as shown in Eq. 4.45 below.

\[
\left( \frac{\partial \rho}{\partial i} \right)_{i,j}^{n} \frac{i_{j} - i_{i,j}}{\Delta t} + \left( \frac{\partial \rho}{\partial p} \right)_{i,j}^{n} \frac{p_{i,j} - p_{i,j}^{n}}{\Delta t} \\
+ \frac{\rho_{o}^{u} u_{B} - \rho_{i,j}^{u} u_{T}}{\Delta x} + \rho_{i,j}^{v} \frac{v_{o} - v_{j}}{\Delta r} + v_{i,j} \left( \frac{\rho_{o} - \rho_{L}}{\Delta r} + \frac{\rho_{i,j}^{p}}{r_{i,j}} \right) = -\dot{i}_{i,j}^{\text{mass}} \left( i_{j,i,j}^{n} - i_{i,j} \right)
\] (4.44)

In the next several equations, the general formulation of the mass transfer term will be used to reduce unnecessary duplications.

The expression for the new time axial velocity that is used is that from Eq. 4.16 above. The new time radial velocity is given in Eq. 4.33. When the expressions for the axial and radial velocities are substituted into Eq. 4.42 the continuity equation becomes Eq. 4.46 below.
In Eq. 4.46 above, the indexing of the velocities is based on the two momentum cell meshes. The continuity cell values are indexed on the continuity cells and the asterisked terms are donor cell values which depend on the value of the velocities. In Eq. 4.47 below, the explicit terms are taken to the right hand side of the continuity equation.
The terms on the left hand side are grouped by their new time quantity, either pressure or enthalpy, and all terms are multiplied by $r_{i,j}\Delta t\Delta x\Delta r$ in Eq. 4.48. This equation also shows a non-interface cell in which there would be no mass transfer.
\[
\begin{align*}
&\left[ \frac{\partial \rho}{\partial t} \right]_{i,j}^{n} r_{i,j} \Delta x \Delta r + \left( c_{i+1,j}^{2} \rho_{B}^{*} - c_{i,j}^{2} \rho_{i}^{*} \right) r_{i,j} \Delta t \Delta r \\
&+ \left( d_{i+1,j}^{2} - d_{i,j}^{2} \right) r_{i,j} \rho_{i,j} \Delta t \Delta x \\
&+ \frac{1}{2} \left( \rho_{i,j} \Delta r + \left[ \rho_{o} - \rho_{i} \right] r_{i,j} \right) \left( d_{i,j}^{2} + d_{i+1,j}^{2} \right) \Delta t \Delta x
\end{align*}
\]
At the vapor side of the interface, Eq. 4.48 becomes Eq. 4.49 given below.

\[
\left\{ \frac{\partial \rho}{\partial t} \right\}_{i,j} \left[ r_{i,j} \Delta x \Delta r + \left( c_{i+1,j}^2 \rho_b^* - c_{i,j}^2 \rho_r^* \right) r_{i,j} \Delta t \Delta r \right] \\
i_{i,j} [d_{i,j+1}^2 - d_{i,j}^2] r_{i,j} \rho_{i,j} \Delta t \Delta x \\
+ \frac{1}{2} \left[ \rho_{i,j} \Delta r + \left[ \rho_o - \rho_r \right] r_{i,j} \right] \left( d_{i,j}^2 + d_{i,j+1}^2 \right) \Delta t \Delta x \\
+ p_{i,j} \left[ c_{i+1,j}^2 \rho_b^* r_{i,j} \Delta t \Delta r \right] \\
+ p_{i+1,j} \left( c_{i+1,j}^0 + c_{i,j}^1 \right) \rho_b^* r_{i,j} \Delta t \Delta r \\
-i_{i,j} \left[ c_{i,j}^2 \rho_r^* r_{i,j} \Delta t \Delta r \right] \\
+ p_{i-1,j} \left[ c_{i,j}^0 - c_{i-1,j}^1 \right] \rho_r^* r_{i,j} \Delta t \Delta r \\
+i_{i,j+1} \left[ d_{i,j+1}^2 \left( \rho_{i,j} r_{i,j} + \frac{1}{2} \left( \rho_{i,j} \Delta r + \left[ \rho_o - \rho_r \right] r_{i,j} \right) \right) \Delta t \Delta x \right] \\
+ p_{i,j+1} \left[ \left( d_{i,j+1}^0 + d_{i,j+1}^1 \right) \rho_{i,j} r_{i,j} \Delta t \Delta x \right] \\
+ \frac{1}{2} \left( d_{i,j+1}^0 + d_{i,j+1}^1 \right) \left( \rho_{i,j} \Delta r + \left[ \rho_o - \rho_r \right] r_{i,j} \right) \Delta t \Delta x \\
+i_{i,j-1} \left[ d_{i,j}^2 \left( \frac{1}{2} \left( \rho_{i,j} \Delta r + \left[ \rho_o - \rho_r \right] r_{i,j} \right) - r_{i,j} r_{i,j} \right) \Delta t \Delta x + \hat{\Gamma}_{i,j} \rho_{i,j} \Delta t \Delta x \Delta r + \right] \\
+ p_{i,j-1} \left[ \left( d_{i,j}^0 - d_{i,j}^1 \right) \rho_{i,j} r_{i,j} \Delta t \Delta x + \frac{1}{2} \left( d_{i,j}^1 - d_{i,j}^0 \right) \left( \rho_{i,j} \Delta r + \left[ \rho_o - \rho_r \right] r_{i,j} \right) \Delta t \Delta x \right] \\
= \hat{\Gamma}_{i,j} \rho_{i,j} \Delta t \Delta x + \frac{1}{2} \left( \rho_{i,j} \Delta r + \left[ \rho_o - \rho_r \right] r_{i,j} \right) \left( v_{i,j}^0 + v_{i,j+1}^0 \right) \Delta t \Delta x \\
+ \frac{1}{2} \left( \rho_{i,j} \Delta r + \left[ \rho_o - \rho_r \right] r_{i,j} \right) \left( v_{i,j}^0 - v_{i,j-1}^0 \right) \Delta t \Delta x
At the liquid side of the interface, Eq. 4.48 becomes Eq. 4.50 below.

\[
\begin{align*}
    \left[ \frac{\partial \rho}{\partial t} \right]_{i,j} & = r_{i,j} \Delta x \Delta r + \left( c_{i+1,j}^2 \rho_B^* - c_{i,j}^2 \rho_T^* \right) r_{i,j} \Delta t \Delta r - \Gamma_{i,j}^{\omega} r_{i,j} \Delta t \Delta x \\
    & + \left( d_{i,j+1}^2 - d_{i,j}^2 \right) r_{i,j} \rho_{i,j} \Delta t \Delta x + \frac{1}{2} \left( \rho_{o} - \rho_{t} \right) r_{i,j} \left( d_{i,j}^2 + d_{i,j+1}^2 \right) \Delta t \Delta x \\
    & + \left[ \frac{\partial \rho}{\partial p} \right]_{i,j} r_{i,j} \Delta t \Delta r + \left\{ \left( c_{i+1,j}^1 - c_{i,j}^0 \right) \rho_B^* - \left( c_{i,j}^0 + c_{i+1,j}^1 \right) \rho_T^* \right\} r_{i,j} \Delta t \Delta r \\
    & + p_{i,j} + \left\{ \left( d_{i,j+1} - d_{i,j} \right) - \left( d_{i,j}^0 + d_{i+1,j}^0 \right) \right\} r_{i,j} \rho_{i,j} \Delta t \Delta x \\
    & + \frac{1}{2} \left( d_{i,j}^0 + d_{i,j+1}^0 - d_{i,j+1}^0 + d_{i,j}^1 \right) \left( \rho_{i,j} \Delta r + \left[ \rho_{o} - \rho_{t} \right] r_{i,j} \right) \Delta t \Delta x \\
    & + i_{i+1,j} \left[ c_{i+1,j}^2 \rho_B^* r_{i,j} \Delta t \Delta r \right] \\
    & + p_{i+1,j} \left[ c_{i+1,j}^0 + c_{i,j+1}^1 \right] \rho_B^* r_{i,j} \Delta t \Delta r \\
    & - i_{i-1,j} \left[ c_{i,j}^2 \rho_T^* r_{i,j} \Delta t \Delta r \right] \\
    & + p_{i-1,j} \left[ \left( c_{i,j}^0 - c_{i-1,j}^1 \right) \rho_T^* r_{i,j} \Delta t \Delta r \right] \\
    & + i_{i,j+1} \left[ d_{i,j+1}^2 \left( \rho_{i,j} r_{i,j} + \frac{1}{2} \left( \rho_{o} - \rho_{t} \right) \right) \Delta t \Delta x \right] \\
    & + p_{i,j+1} \left[ \left( d_{i,j+1}^0 + d_{i+1,j}^1 \right) \rho_{i,j} r_{i,j} \Delta t \Delta x + \frac{\partial i_{i,j+1} \Gamma_{i,j}^{\omega} r_{i,j} \Delta t \Delta x}{\partial p} \right] \\
    & + \frac{1}{2} \left( d_{i,j+1}^0 + d_{i+1,j}^1 \right) \left( \rho_{i,j} \Delta r + \left[ \rho_{o} - \rho_{t} \right] r_{i,j} \right) \Delta t \Delta x \\
    & + i_{i-1,j} \left[ d_{i,j}^2 \left( \frac{1}{2} \left( \rho_{o} - \rho_{t} \right) \right) - \rho_{i,j} r_{i,j} \right) \Delta t \Delta x \right] \\
    & + p_{i,j-1} \left[ \left( d_{i,j}^0 - d_{i,j}^1 \right) \rho_{i,j} r_{i,j} \Delta t \Delta x + \frac{1}{2} \left( d_{i,j}^1 - d_{i,j}^0 \right) \left( \rho_{i,j} \Delta r + \left[ \rho_{o} - \rho_{t} \right] r_{i,j} \right) \Delta t \Delta x \right] \\
    & = \Gamma_{i,j}^{\omega} \left[ p_{i+1,j} \frac{\partial i_{i,j+1}}{\partial p} - i_{i,j+1}^0 \right] r_{i,j} \Delta t \Delta x + \left[ \left( \frac{\partial \rho}{\partial t} \right)_{i,j} r_{i,j} + \left( \frac{\partial \rho}{\partial p} \right)_{i,j} \right] \Delta t \Delta x + r_{i,j} \Delta t \Delta r \\
    & + \left[ \rho_T^* u_{i,j}^0 - \rho_B^* u_{i+1,j}^0 \right] r_{i,j} \Delta t \Delta r + \left[ v_{i,j}^0 - v_{i+1,j}^0 \right] r_{i,j} \rho_{i,j} \Delta t \Delta x \\
    & - \frac{1}{2} \left( \rho_{o} - \rho_{t} \right) r_{i,j} \Delta t \Delta x \\
    & + \left[ \rho_T^* u_{i,j}^0 - \rho_B^* u_{i+1,j}^0 \right] r_{i,j} \Delta t \Delta r + \left[ v_{i,j}^0 - v_{i+1,j}^0 \right] r_{i,j} \rho_{i,j} \Delta t \Delta x \\
    & - \frac{1}{2} \left( \rho_{o} - \rho_{t} \right) r_{i,j} \Delta t \Delta x
\end{align*}
\]
4.4 Energy Equation Linearization

The energy equation, like the momentum equations and the continuity equation is solved in a semi-implicit method. Like the continuity equation the only terms that are new time values, other than in the time derivative, are the velocities in the convection terms and the heat and mass transfer terms. The two-dimensional energy equation in cylindrical coordinates is given in Eq. 4.51.

\[
\frac{\partial \rho_i}{\partial t} + \frac{\partial \rho_i u}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} \left( \rho_i v r \right) = \dot{\Gamma}_i + q_i + \frac{\partial p}{\partial t} + \rho \varepsilon_i \left( \frac{\partial^2 i}{\partial x^2} + \frac{\partial^2 i}{\partial r^2} + \frac{1}{r} \frac{\partial i}{\partial r} \right) \tag{4.51}
\]

The time derivative in the energy equation is expanded by product rule in Eq. 4.52

\[
\frac{\partial \rho_i}{\partial t} = \rho \frac{\partial i}{\partial t} + i \frac{\partial \rho}{\partial t} \tag{4.52}
\]

The density is a function of both pressure and enthalpy. The time derivative of the density can be expanded as in Eq. 4.53

\[
\frac{\partial \rho_i}{\partial t} = \rho \frac{\partial i}{\partial t} + i \left[ \frac{\partial \rho}{\partial t} \frac{\partial i}{\partial \rho} + \frac{\partial \rho}{\partial p} \frac{\partial p}{\partial t} \right] \tag{4.53}
\]

A finite volume formulation of the time derivative in the energy equation is given in Eq. 4.54

\[
\frac{\partial \rho_i}{\partial t} = \left[ \dot{i}_{i,j} \left( \frac{\partial \rho}{\partial i} \right) + \rho_{i,j} \right] + \frac{\rho_{i,j} - \dot{i}_{i,j} + \dot{i}_{i,j} \left( \frac{\partial \rho}{\partial p} \right)}{\Delta t} + \frac{p_{i,j} - p_n}{\Delta t} \tag{4.54}
\]
In the time derivative of the energy equation above, the derivatives of density with respect to enthalpy and pressure are determined from the equation of state and used as old time values.

The finite volume formulation of the energy equation axial flux term is given in Eq. 4.55

\[
\frac{\partial \rho i u}{\partial x} = \frac{(\rho i)_B^{*} u_B - (\rho i)_T^{*} u_T}{\Delta x}
\]  

(4.55)

In the above equation, the subscript \( B \) refers to the bottom, or \( i + \frac{1}{2} \), face and the subscript \( T \) refers to the top, or \( i - \frac{1}{2} \), face. The superscript asterisk designates the terms being fluxed by the velocities. These are donor cell quantities. The velocities are taken at the computational cell axial faces, which is the center of the axial momentum cell.

The radial convection term in the energy equation is expanded using product rule in Eq. 4.56

\[
\frac{1}{r} \frac{\partial}{\partial r} [\rho i v r] = \rho i \frac{\partial v}{\partial r} + \rho v \frac{\partial i}{\partial r} + iv \frac{\partial \rho}{\partial r} + \frac{\rho iv}{r}
\]  

(4.56)

The finite volume formulation of the radial flux term is written in Eq. 4.57

\[
\frac{1}{r} \frac{\partial}{\partial r} [\rho i v r] = \rho_{i,j} v_{i,j} \frac{v_o - v_i}{\Delta r} + \left( \rho_{i,j} \frac{i_o^n - i_i^n}{\Delta r} + \rho_o - \rho_i + \rho_{i,j} \frac{i_o^n}{r_{i,j}} \right) v_{i,j}
\]  

(4.57)
In the above finite volume formulation, the subscript $O$ refers to the outside, $j + \frac{1}{2}$, face of the continuity cell and the subscript $I$ refers to the inside, $j - \frac{1}{2}$, face. The superscript asterisk designates the donor cell quantities being fluxed by the velocity. The velocities at the inside and outside faces are the cell center velocities in the radial momentum cells. In the second term of Eq. 4.57 all of the values are cell centered. The cell centered velocity is found by linear average of the two radial face velocities. The radius is the radius at the center of the continuity cell.

The finite volume formulation of the energy source from mass transfer term is given in Eq. 4.58

$$\Gamma^{*\epsilon}_i = \Gamma^{*\epsilon}_{i,j} (i_j - i_l) l^\epsilon_{i,j}, \text{ where } \Gamma^{*\epsilon}_{i,j} = \frac{h_{ij} A_{HT}}{i_{fs} c_p V_{i,j}}$$

(4.58)

In the above equation, $h_{ij}$ is the liquid side heat transfer coefficient, $A_{HT}$ is the heat transfer area, $c_p$ is the specific heat at constant pressure, $V_{i,j}$ is the cell volume, and $i_{fs}^*$ is the latent heat of vaporization. All of these terms are taken at the old time value. The new time enthalpies are given at the interface and the surface of the liquid jet. The mass transfer rate is a cell centered value given in the continuity cell. The mass transfer enthalpy is dependant on the location and situation. In the vapor boundary layer, in the case of condensation, the mass transfer enthalpy is equal to the enthalpy within the continuity cell. In the liquid jet, during condensation, the mass transfer enthalpy is equal
to the liquid saturation enthalpy. On the vapor side of the interface, the mass transfer term takes the form of Eq. 4.59.

\[ \Gamma_{i,j}^\text{mr} = \Gamma_{i,j}^\text{mr} n_i \left( i_{f,i,j} - i_{i,j-1} \right) \]  \hspace{1cm} (4.59)

Given in Eq. 4.60 is the mass transfer term on the liquid side of the interface. On the liquid side of the interface, the mass transfer term is also subtracted from the right hand side instead of added to as in the vapor terms.

\[ \Gamma_{i,j} = \Gamma_{i,j}^\text{mr} n_i \left( i_{f,i,j+1} - i_{i,j} \right) \]  \hspace{1cm} (4.60)

The heat transfer term, because it is assumed that the vapor is saturated, only exists on the liquid side of the interface. The implicit form of the liquid side heat transfer term is given in Eq. 4.61 below.

\[ q_{li,i,j}^\text{mr} = q_{li,i,j}^\text{mr} \left( i_{f,i,j+1} - i_{i,j} \right), \text{ where } q_{li,i,j}^\text{mr} = \frac{h_d A_{HT}}{i_{f,i} c_p V_{i,j}} \]  \hspace{1cm} (4.61)

In the three previous equations the liquid saturation is taken to be at the new time value. However at the time of calculation, the fluid properties are only known at the old time value. The derivative of the saturation enthalpy with pressure is also known and can be used to find the new time value as shown in Eq. 4.62 below.

\[ i_{f,i,j} = i_{f,i,j}^n + \frac{\partial i_{f,i,j}}{\partial p} \left( p_{i,j} - p_{i,j}^n \right) \]  \hspace{1cm} (4.62)
The finite volume formulation of the pressure work term is given in Eq. 4.63

\[
\frac{\partial p}{\partial t} = \frac{p_{i,j} - p_{i,j}^n}{\Delta t}
\]  

(4.63)

In the pressure work term, the pressures, both new and old time, are cell centered quantities.

The thermal diffusion term in the energy equation is given, in its finite volume formulation, in Eq. 4.64

\[
\rho \varepsilon_H \left[ \frac{\partial^2 i}{\partial x^2} + \frac{\partial^2 i}{\partial r^2} + \frac{1}{r} \frac{\partial i}{\partial r} \right] = \rho_{i,j} \varepsilon_{H,i,j} \left[ \frac{i_{i+1,j}^n - 2i_{i,j}^n + i_{i-1,j}^n}{\Delta x^2} + \frac{i_{i,j+1}^n - 2i_{i,j}^n + i_{i,j-1}^n}{\Delta r^2} + \frac{i_{i,j+1}^n - i_{i,j-1}^n}{2r_{i,j} \Delta r} \right]
\]  

(4.64)

The thermal diffusivity, \( \varepsilon_H \), depending on the flow conditions can be either the molecular thermal diffusivity of the fluid or the turbulent thermal diffusivity, which is given from the turbulent momentum diffusivity and the turbulent Prandtl number. The density and the enthalpies in each of the terms is a cell centered value.

The finite volume formulation of the full energy equation at a non-interface location is given below in Eq. 4.65
The energy equation at the vapor side of the interface is given in Eq. 4.66 below.

\[
\left[ i_{i,j}^n \left( \frac{\partial \rho}{\partial t} \right)_{i,j}^n + \rho_{i,j} \right] \frac{i_{i,j}^n - i_{i,j}^{n-1}}{\Delta t} + i_{i,j}^n \left( \frac{\partial \rho}{\partial p} \right)_{i,j}^n \frac{p_{i,j} - p_{i,j}^n}{\Delta t} + \left( \rho i \right)_T^* u_b - \left( \rho i \right)_T^* u_T \\
+ \rho_{i,j} i_{i,j}^n \frac{v_o - v_T}{\Delta r} + \left( \rho_{i,j} \frac{i_{i,j}^n - i_{i,j}^n}{\Delta r} + i_{i,j}^n \frac{\rho_{i,j} - \rho_{i,j}}{r_{i,j}} + \rho_{i,j} i_{i,j}^n \right) v_{i,j} \\
= \frac{p_{i,j} - p_{i,j}^n}{\Delta t} \\
+ \rho_{i,j} \epsilon_{H_{i,j}} \left[ \frac{i_{i+j}^{n+1} - 2i_{i,j}^n + i_{i-j,2}^n}{\Delta x^2} + \frac{i_{i,j}^{n+1} - 2i_{i,j}^n + i_{i,j-2}^n}{\Delta r^2} + \frac{i_{i,j}^{n+1} - i_{i-j}^n}{2r_{i,j} \Delta r} \right]
\] (4.66)

On the liquid side of the interface the energy equation is given by Eq. 4.67.

\[
\left[ i_{i,j}^n \left( \frac{\partial \rho}{\partial t} \right)_{i,j}^n + \rho_{i,j} \right] \frac{i_{i,j}^n - i_{i,j}^{n-1}}{\Delta t} + i_{i,j}^n \left( \frac{\partial \rho}{\partial p} \right)_{i,j}^n \frac{p_{i,j} - p_{i,j}^n}{\Delta t} + \left( \rho i \right)_T^* u_b - \left( \rho i \right)_T^* u_T \\
+ \rho_{i,j} i_{i,j}^n \frac{v_o - v_T}{\Delta r} + \left( \rho_{i,j} \frac{i_{i,j}^n - i_{i,j}^n}{\Delta r} + i_{i,j}^n \frac{\rho_{i,j} - \rho_{i,j}}{r_{i,j}} + \rho_{i,j} i_{i,j}^n \right) v_{i,j} \\
= \left\{ \gamma_{i,j}^{\text{out}} \right\}_{i,j} \left( \frac{i_{i+j}^{n+1} - i_{i,j}^n}{\Delta t} \\
+ \rho_{i,j} \epsilon_{H_{i,j}} \left[ \frac{i_{i+j}^{n+1} - 2i_{i,j}^n + i_{i-j,2}^n}{\Delta x^2} + \frac{i_{i,j}^{n+1} - 2i_{i,j}^n + i_{i,j-2}^n}{\Delta r^2} + \frac{i_{i,j}^{n+1} - i_{i-j}^n}{2r_{i,j} \Delta r} \right]\right\}
\] (4.67)
The expression for the new time axial velocity that is used is that from Eq. 4.16 above. The new time radial velocity is given in Eq. 4.33. When the expressions for the axial and radial velocities are substituted into Eq. 4.65 the energy equation becomes Eq. 4.68 below. The next several steps in the linearization process will be shown only at the non-interface location. The final equation will be shown at all locations.

\[
\begin{align*}
\left[i_{i,j}^n \left( \frac{\partial p}{\partial t} \right)_{i,j}^n + \rho_{i,j} \right] & \frac{i_{i,j} - i_{i,j}^n}{\Delta t} + \left[ \frac{\partial \rho}{\partial t} \right]_{i,j}^n \frac{p_{i,j} - p_{i,j}^n}{\Delta t} \\
+ \left( \frac{\rho \rho}{\Delta x} \right)_B & \left[ u_{i+1,j}^0 + c_{i+1,j}^0 \left( p_{i+1,j} - p_{i,j} \right) + c_{i+1,j}^1 \left( p_{i+1,j} + p_{i,j} \right) + c_{i+1,j}^2 \left[i_{i+1,j} + i_{i,j} \right] \right] \\
- \left( \frac{\rho \rho}{\Delta x} \right)_T & \left[ u_{i,j}^0 + c_{i,j}^0 \left( p_{i,j} - p_{i-1,j} \right) + c_{i,j}^1 \left( p_{i,j} + p_{i-1,j} \right) + c_{i,j}^2 \left[i_{i,j} + i_{i-1,j} \right] \right] \\
+ \frac{\rho_{i,j}}{\Delta r} & \left[ v_{i+1,j}^0 + d_{i+1,j}^0 \left( p_{i,j+1} - p_{i,j} \right) + d_{i+1,j}^1 \left( p_{i,j+1} + p_{i,j} \right) + d_{i+1,j}^2 \left[i_{i,j+1} + i_{i,j} \right] \right] \\
- \frac{\rho_{i,j}}{\Delta r} & \left[ v_{i,j}^0 + d_{i,j}^0 \left( p_{i,j} - p_{i-1,j} \right) + d_{i,j}^1 \left( p_{i,j} + p_{i-1,j} \right) + d_{i,j}^2 \left[i_{i,j} + i_{i-1,j} \right] \right] \\
+ \frac{1}{2} \left( \rho_{i,j} \frac{i_{i,j}^0 - i_{i,j}^n}{\Delta r} + \rho_{i,j} \frac{\rho_{i,j} - \rho_L}{\Delta r} + \frac{\rho_{i,j}}{\Delta r} \right) & \left[ v_{i,j}^0 + d_{i,j}^0 \left( p_{i,j} - p_{i,j-1} \right) + d_{i,j}^1 \left( p_{i,j} + p_{i,j-1} \right) + d_{i,j}^2 \left[i_{i,j} + i_{i,j-1} \right] \right] \\
+ \frac{1}{2} \left( \rho_{i,j} \frac{i_{i,j}^0 - i_{i,j}^n}{\Delta r} + \frac{\rho_{i,j}}{\Delta r} \right) & \left[ v_{i,j+1}^0 + d_{i,j+1}^0 \left( p_{i,j+1} - p_{i,j} \right) + d_{i,j+1}^1 \left( p_{i,j+1} + p_{i,j} \right) + d_{i,j+1}^2 \left[i_{i,j+1} + i_{i,j} \right] \right] \\
= & \frac{p_{i,j} - p_{i,j}^n}{\Delta t} \\
+ \rho_{i,j} \mathcal{E}_{i,j} & \left[ \frac{i_{i,j}^n - i_{i,j}^n - 2i_{i,j} + i_{i,j+1}^n - 2i_{i,j}^n + i_{i,j+1}^n - i_{i,j-1}^n}{\Delta x^2} + \frac{i_{i,j+1}^n - i_{i,j+1}^n - 2i_{i,j}^n + i_{i,j+1}^n + i_{i,j+1}^n - i_{i,j-1}^n}{\Delta r^2} + \frac{2r_{i,j}}{\Delta r} \right]
\end{align*}
\]
In Eq. 4.69 below, the explicit terms are moved to the right hand side and all terms are multiplied by \( r_{i,j} \Delta t \Delta x^2 \Delta r^2 \).

\[
\begin{align*}
&\left\{\left[i^n_{i,j \partial_{\rho \partial t}^n}_{i,j} + \rho_{i,j}\right]_{i,j} + \left[i^n_{i,j \partial_{\rho \partial p}^n}_{i,j} - 1\right]_{i,j}\right\} r_{i,j} \Delta x^2 \Delta r^2 \\
&\quad + \left(\rho i\right)_B r_{i,j} \Delta t \Delta x \Delta r^2 \left[ + e_{i+1,j}^0 \left[p_{i+1,j} - p_{i,j}\right] - e_{i,j}^1 \left[p_{i,j} + p_{i-1,j}\right] + e_{i+1,j}^2 \left[i_{i+1,j} + i_{i,j}\right]\right] \\
&\quad - \left(\rho i\right)_T r_{i,j} \Delta t \Delta x \Delta r^2 \left[ + e_{i,j}^0 \left[p_{i,j} - p_{i-1,j}\right] + e_{i,j}^1 \left[p_{i,j} + p_{i-1,j}\right] + e_{i,j}^2 \left[i_{i,j} + i_{i-1,j}\right]\right] \\
&\quad + \rho_{i,j} \left(i^n_{i,j} - i^n_{i,j}\right) \Delta t \Delta x^2 \Delta r \left[ + i^n_{i,j} \left(p_{i,j} + p_{i,j-1}\right) + d^n_{i+1,j} \left[p_{i,j} + p_{i,j-1}\right]\right] \\
&\quad + \left[\rho_{i,j} \left(i^n_{i,j} \left(p_{i,j} - p_{i,j-1}\right) + \rho_{i,j} \left(i^n_{i,j} \Delta r\right)\right) + \left(i^n_{i,j} \rho_{i,j} - \rho_{i,j}\right) + \rho_{i,j} \left(i^n_{i,j} \Delta r\right)\right] / 2 \left[ + i^n_{i,j} \left[p_{i,j} + p_{i,j-1}\right]\right] \\
&\quad = \left\{\left[i^n_{i,j \partial_{\rho \partial t}^n}_{i,j} + \rho_{i,j}\right]_{i,j} + \left[i^n_{i,j \partial_{\rho \partial p}^n}_{i,j} - 1\right]_{i,j}\right\} r_{i,j} \Delta x^2 \Delta r^2 \\
&\quad + \left(u^0_{i,j} \left(\rho i\right)_T - u^0_{i+1,j} \left(\rho i\right)_B\right) r_{i,j} \Delta t \Delta x \Delta r^2 \\
&\quad + \left(v^0_{i,j} - v^0_{i+1,j}\right) r_{i,j} \rho_{i,j} \left(i^n_{i,j} \Delta r\right) \Delta t \Delta x^2 \Delta r \\
&\quad - \frac{1}{2} \left(\rho_{i,j} \left(i^n_{i,j} - i^n_{i,j}\right) + i^n_{i,j} \left(\rho_{i,j} - \rho_{i,j}\right) + \rho_{i,j} \left(i^n_{i,j} \Delta r\right)\right) \left[v^0_{i,j} + v^0_{i+1,j}\right] \Delta t \Delta x^2 \Delta r \\
&\quad + \rho_{i,j} \Delta t \varepsilon_{H,i,j} \left[ \left(i^n_{i,j} - 2i^n_{i,j} + i^n_{i,j-1}\right) r_{i,j} \Delta r^2 + \left(i^n_{i,j} - 2i^n_{i,j} + i^n_{i,j-1}\right) r_{i,j} \Delta x^2 \right] \\
&\quad + 0.5 \left(i^n_{i,j+1} - i^n_{i,j-1}\right) \Delta x^2 \Delta r \quad (4.69)
\end{align*}
\]

Finally, the terms on the left hand side of the equation are grouped according to the implicit variable, either enthalpy or pressure in Eq. 4.70.
\[
\begin{align*}
&\left( i_{i,j}^n \left( \frac{\partial \rho}{\partial t} \right)_{i,j} + \rho_{i,j} \right) r_{i,j} \Delta x^2 \Delta r^2 \\
&+ \left( c_{i+1,j}^2 \left( \rho i \right)_B - c_{i,j}^2 \left( \rho i \right)_T \right) r_{i,j} \Delta t \Delta x \Delta r^2 + \left( d_{i+1,j}^2 - d_{i,j}^2 \right) r_{i,j} \rho_i \rho_{i,j} \Delta t \Delta x^2 \Delta r \\
&+ \frac{1}{2} \left( d_{i,j}^2 + d_{i+1,j}^2 \right) \left( \rho_{i,j} r_{i,j} \left( i_0^n - i_1^n \right) + i_{i,j}^n \rho \Delta t \Delta x^2 \Delta r \right)
\end{align*}
\]

\[
\begin{align*}
&\left( i_{i,j}^n \left( \frac{\partial \rho}{\partial p} \right)_{i,j} -1 \right) r_{i,j} \Delta x^2 \Delta r^2 \\
&+ \left\{ \left( c_{i+1,j}^1 - c_{i,j}^0 \right) \left( \rho i \right)_B - \left( c_{i,j}^0 + c_{i,j}^1 \right) \left( \rho i \right)_T \right\} r_{i,j} \Delta t \Delta x \Delta r^2 \\
&+ \left\{ \left( d_{i,j+1}^1 - d_{i,j}^0 \right) - \left( d_{i,j+1}^0 + d_{i,j}^1 \right) \right\} r_{i,j} \rho_i \rho_{i,j} \Delta t \Delta x^2 \Delta r \\
&+ \frac{1}{2} \left( d_{i,j}^0 + d_{i,j+1}^1 - d_{i,j}^0 + d_{i,j+1}^1 \right) \left( \rho_{i,j} r_{i,j} \left( i_0^n - i_1^n \right) + i_{i,j}^n \rho \Delta t \Delta x^2 \Delta r \right)
\end{align*}
\]

\[
\begin{align*}
&+ i_{i+1,j} \left( c_{i+1,j}^2 \left( \rho i \right)_B r_{i,j} \Delta t \Delta x \Delta r^2 \right) \\
&+ p_{i,j+1} \left[ \left( c_{i+1,j}^0 + c_{i,j}^1 \right) \left( \rho i \right)_T r_{i,j} \Delta t \Delta x \Delta r^2 \right] \\
&- i_{i-1,j} \left( c_{i,j}^2 \left( \rho i \right)_T r_{i,j} \Delta t \Delta x \Delta r^2 \right) \\
&+ p_{i,j-1} \left[ \left( c_{i,j}^0 - c_{i,j}^1 \right) \left( \rho i \right)_T r_{i,j} \Delta t \Delta x \Delta r^2 \right]
\end{align*}
\]

\[
\begin{align*}
&\left[ d_{i,j+1}^2 \left\{ \rho_{i,j} r_{i,j} \Delta t \Delta x^2 \Delta r + \frac{1}{2} \left( \rho_{i,j} r_{i,j} \left( i_0^n - i_1^n \right) + i_{i,j}^n \rho \right) \right\} \Delta t \Delta x^2 \Delta r \right] \\
&\left[ \left( d_{i,j+1}^0 + d_{i,j}^1 \right) \left\{ \frac{1}{2} \left( \rho_{i,j} r_{i,j} \left( i_0^n - i_1^n \right) + i_{i,j}^n \rho \right) \right\} \Delta t \Delta x^2 \Delta r \right] \\
&\left[ i_{i,j+1} \left\{ d_{i,j+1}^2 \left\{ \frac{1}{2} \left( \rho_{i,j} r_{i,j} \left( i_0^n - i_1^n \right) + i_{i,j}^n \rho \right) \right\} \Delta t \Delta x^2 \Delta r \right] \\
&\left[ p_{i,j+1} \left\{ \left( d_{i,j+1}^1 - d_{i,j}^0 \right) \left\{ \frac{1}{2} \left( \rho_{i,j} r_{i,j} \left( i_0^n - i_1^n \right) + i_{i,j}^n \rho \right) \right\} \Delta t \Delta x^2 \Delta r \right] \\
&\left[ - p_{i,j+1} \left\{ \left( d_{i,j+1}^0 - d_{i,j}^1 \right) \left\{ \frac{1}{2} \left( \rho_{i,j} r_{i,j} \left( i_0^n - i_1^n \right) + i_{i,j}^n \rho \right) \right\} \Delta t \Delta x^2 \Delta r \right]
\end{align*}
\]
\[
\begin{align*}
&= \left\{ i_{t,j}^{n} \left( \frac{\partial \rho}{\partial t} \right)_{t,j} + \rho_{i,j} \right\} i_{t,j}^{n} + \left[ i_{t,j}^{n} \left( \frac{\partial \rho}{\partial t} \right)_{i,j} - 1 \right] p_{i,j}^{n} \right\} r_{i,j} \Delta x^2 \Delta r^2 \\
&+ \left( u_{i,j}^{0} (\rho i)^{0}_r - u_{i+1,j}^{0} (\rho i)^{0}_B \right) r_{i,j} \Delta t \Delta x \Delta r^2 \\
&+ \left( v_{i,j}^{0} - v_{i,j+1}^{0} \right) r_{i,j} \rho_{i,j} i_{i,j}^{n} \Delta t \Delta x^2 \Delta r \\
&- \frac{1}{2} \left( \rho_{i,j} r_{i,j} \left( i_{0}^{n} - i_{t}^{n} \right) + i_{i,j}^{n} r_{i,j} \left( \rho_{0} - \rho_{t} \right) + \rho_{i,j} i_{i,j}^{n} \Delta r \right) \left( v_{i,j}^{0} + v_{i,j+1}^{0} \right) \Delta t \Delta x^2 \Delta r \\
&+ \rho_{i,j} \Delta t \epsilon_H_{i,j} \left[ \left( i_{t,j+1}^{n} - 2i_{i,j}^{n} + i_{i-1,j}^{n} \right) r_{i,j} \Delta r^2 + \left( i_{i,j+1}^{n} - 2i_{i,j}^{n} + i_{i,j-1}^{n} \right) r_{i,j} \Delta x^2 \right] \\
&+ 0.5 \left( i_{i,j+1}^{n} - i_{i,j-1}^{n} \right) \Delta x^2 \Delta r
\end{align*}
\]

On the vapor side if the liquid jet/vapor interface, the energy equation is written as shown in Eq. 4.71 below.
\[
\begin{align*}
&\left[ \left( i_{i,j}^n \left( \frac{\partial \rho}{\partial i} \right)_{i,j}^n + \rho_{i,j} \right) r_{i,j} \Delta x^2 \Delta r^2 \right] \\
&+ \left( c_{i+1,j}^2 \left( \rho_i \right)_{i,j}^n - c_{i,j}^2 \left( \rho_i \right)_{i,j}^n \right) r_{i,j} \Delta t \Delta x \Delta r^2 + \left( d_{i,j+1}^2 - d_{i,j}^2 \right) r_{i,j} \rho_{i,j} \Delta t \Delta x^2 \Delta r \\
&+ \frac{1}{2} \left( d_{i,j+1}^2 + d_{i,j}^2 \right) \left( \rho_{i,j} \Delta r \left( i_{i,j}^n - i_{i+1,j}^n \right) + i_{i,j}^n \rho_{i,j} \Delta r \left( \rho_o - \rho_i \right) \right) \Delta t \Delta x^2 \Delta r \\
&+ \left[ \left( i_{i,j}^n \left( \frac{\partial \rho}{\partial p} \right)_{i,j}^n - 1 \right) r_{i,j} \Delta x^2 \Delta r^2 - i_{i,j}^n \rho_{i,j} \frac{\partial i_{i,j}^n}{\partial p} r_{i,j} \Delta t \Delta x \Delta r^2 \right] \\
&+ \left\{ \left( c_{i+1,j}^2 - c_{i,j}^2 \right) \left( \rho_i \right)_{i,j}^n - \left( c_{i,j}^2 + c_{i+1,j}^2 \right) \left( \rho_i \right)_{i,j}^n \right\} r_{i,j} \Delta t \Delta x \Delta r^2 \\
&+ \left\{ \left( d_{i,j+1}^2 - d_{i,j}^2 \right) - \left( d_{i,j}^2 + d_{i,j+1}^2 \right) \right\} r_{i,j} \rho_{i,j} \Delta t \Delta x^2 \Delta r \\
&+ \frac{1}{2} \left( d_{i,j+1}^2 + d_{i,j}^2 - d_{i,j}^2 + d_{i,j+1}^2 \right) \left( \rho_{i,j} \Delta r \left( i_{i,j}^n - i_{i+1,j}^n \right) + i_{i,j}^n \rho_{i,j} \Delta r \left( \rho_o - \rho_i \right) \right) \Delta t \Delta x^2 \Delta r \\
&+ \left[ \left( c_{i+1,j}^2 \rho_{i} \right)_{i,j}^n \Delta t \Delta x \Delta r^2 \right] \\
&+ \left[ \left( c_{i,j}^2 + c_{i+1,j}^2 \right) \rho_{i,j}^n \Delta t \Delta x \Delta r^2 \right] \\
&- \left[ \left( c_{i,j}^2 - c_{i+1,j}^2 \right) \rho_{i,j}^n \Delta t \Delta x \Delta r^2 \right] \\
&+ \left[ \left( d_{i,j+1}^2 - d_{i,j}^2 \right) \rho_{i,j}^n \Delta t \Delta x \Delta r^2 \right] \\
&+ \left[ \left( d_{i,j+1}^2 + d_{i,j}^2 \right) \rho_{i,j}^n \Delta t \Delta x \Delta r^2 \right] \\
&+ \frac{1}{2} \left( d_{i,j}^2 - d_{i,j+1}^2 \right) \left( \rho_{i,j} \Delta r \left( i_{i,j}^n - i_{i,j+1}^n \right) + i_{i,j+1}^n \rho_{i,j} \Delta r \left( \rho_o - \rho_i \right) \right) \Delta t \Delta x^2 \Delta r \\
&+ \left[ \left( i_{i,j}^n \rho_{i,j} \Delta r \right) \Delta t \Delta x^2 \Delta r^2 \right] \\
&+ \left[ \left( d_{i,j+1}^2 + d_{i,j}^2 \right) \rho_{i,j}^n \Delta t \Delta x \Delta r^2 \right] \\
&+ \frac{1}{2} \left( d_{i,j}^2 - d_{i,j+1}^2 \right) \left( \rho_{i,j} \Delta r \left( i_{i,j}^n - i_{i,j+1}^n \right) + i_{i,j+1}^n \rho_{i,j} \Delta r \left( \rho_o - \rho_i \right) \right) \Delta t \Delta x^2 \Delta r \\
&+ \left[ \left( d_{i,j}^2 - d_{i,j+1}^2 \right) \rho_{i,j}^n \Delta t \Delta x \Delta r^2 \right]
\end{align*}
\]
\[
\begin{align*}
&= \left\{ \left[i^n_{i,j} \left( \frac{\partial \rho}{\partial t} \right)_{i,j} + \rho_{i,j} \right] i^n_{i,j} + \left[i^n_{i,j} \left( \frac{\partial \rho}{\partial p} \right)_{i,j} - 1 \right] p^n_{i,j} \right\} r_{i,j} \Delta x^2 \Delta r^2 \\
&\quad + \Gamma_{i,j} \left[ i^n_{i,j} - p^n_{i,j} \frac{\partial i_{f,j}}{\partial p} \right] r_{i,j} \Delta t \Delta x^2 \Delta r^2 \\
&\quad + \left(u^0_{i,j} (\rho i)^{x}_r - u^0_{i+1,j} (\rho_i)^{x}_B \right) r_{i,j} \Delta t \Delta x \Delta r^2 \\
&\quad + \left(v^0_{i,j} - v^0_{i,j+1} \right) r_{i,j} \rho_{i,j} i^n_{i,j} \Delta t \Delta x^2 \Delta r \\
&\quad - \frac{1}{2} \left[ \rho_{i,j} \left( i^n_{i,j} - i^n_{i,j+1} \right) + i^n_{i,j} \left( \rho_o - \rho_r \right) + \rho_{i,j} i^n_{i,j} \Delta r \right] \left(v^0_{i,j} + v^0_{i,j+1} \right) \Delta t \Delta x^2 \Delta r \\
&\quad + \rho_{i,j} \Delta t \varepsilon_{i,j} \left[ \left(i^n_{i+1,j} - 2i^n_{i,j} + i^n_{i-1,j} \right) r_{i,j} \Delta r^2 + \left(i^n_{i,j+1} - 2i^n_{i,j} + i^n_{i,j-1} \right) r_{i,j} \Delta x^2 \right] \\
&\quad + 0.5 \left(i^n_{i,j+1} - i^n_{i,j-1} \right) \Delta x^2 \Delta r
\end{align*}
\]

The energy equation on the liquid side of the interface is given in Eq. 4.72.
\[
\begin{align*}
&\left( i_{i,j}^n \left( \frac{\partial \rho}{\partial i} \right)_{i,j}^n + \rho_{i,j} \right) r_{i,j} \Delta x^2 \Delta r^2 + \left( \vec{q}_{\vec{u}_{i,j}^n} - \vec{q}_{\vec{u}_{i,j+1}^n} \right) r_{i,j} \Delta t \Delta x^2 \Delta r^2 \\
&+ \left( c_{i+1,j}^2 (\rho i)_B^* - c_{i,j}^2 (\rho i)_T^* \right) r_{i,j} \Delta t \Delta x \Delta r^2 + \left( d_{i,j+1}^2 - d_{i,j}^2 \right) r_{i,j} \rho_{i,j} \rho_{i,j}^n \Delta t \Delta x \Delta r \\
&+ \frac{1}{2} \left( d_{i,j}^2 + d_{i,j+1}^2 \right) \left( \rho_{i,j} r_{i,j} \left( i_0^n - i_i^n \right) + i_{i,j}^n r_{i,j} \left( \rho_o - \rho_1 \right) + \rho_{i,j} i_{i,j}^n \Delta r \right) \Delta t \Delta x^2 \Delta r \\
&+ \left\{ \left( c_{i+1,j}^2 - c_{i,j}^2 \right) (\rho i)_B^* - \left( c_{i,j}^2 + c_{i,j}^1 \right) (\rho i)_T^* \right\} r_{i,j} \Delta t \Delta x^2 \Delta r^2 \\
&+ \left\{ \left( d_{i,j+1} - d_{i,j} \right) - \left( d_{i,j+1}^1 + d_{i,j}^1 \right) \right\} r_{i,j} \rho_{i,j} i_{i,j}^n \Delta t \Delta x \Delta r \\
&+ \frac{1}{2} \left( d_{i,j}^0 + d_{i,j+1} + d_{i,j+1}^1 \right) \left( \rho_{i,j} r_{i,j} \left( i_0^0 - i_i^0 \right) + i_{i,j}^0 r_{i,j} \left( \rho_o - \rho_1 \right) + \rho_{i,j} i_{i,j}^0 \Delta r \right) \Delta t \Delta x^2 \Delta r
\end{align*}
\]

(4.72)
\[
= \left\{ \left[ i_{i,j}^n \left( \frac{\partial \rho}{\partial t} \right)_{i,j} + \rho_{i,j} \right] \right\} i_{i,j}^n + \left[ i_{i,j}^n \left( \frac{\partial \rho}{\partial p} \right)_{i,j} - 1 \right] p_{i,j}^n \right\} r_{i,j} \Delta x^2 \Delta r^2 \\
+ (q_{i,j}^{\pi} - \bar{q}_{i,j}^{\pi} i_{i,j}^n) \left[ i_{i,j+1}^{n} - p_{i,j+1}^{n} \frac{\partial i_{i,j+1}^{n}}{\partial p} \right] r_{i,j} \Delta t \Delta x^2 \Delta r^2 \\
+ \left( u_{i,j}^{0} (\rho i)_{i,j}^{*} - u_{i+1,j}^{0} (\rho i)_{i+1,j}^{*} \right) r_{i,j} \Delta t \Delta x^2 \Delta r \\
+ \left( v_{i,j}^{0} - v_{i+1,j}^{0} \right) r_{i,j} \Delta t \Delta x^2 \Delta r \\
- \frac{1}{2} \left( \rho_{i+1,j} r_{i,j} \left( i_{i,j} - i_{i,j}^{n} \right) + i_{i,j}^{n} r_{i,j} \left( \rho_{i} - \rho_{i,j} \right) + \rho_{i+1,j} i_{i+1,j}^{n} \Delta r \right) \left( v_{i,j}^{0} + v_{i+1,j}^{0} \right) \Delta t \Delta x^2 \Delta r \\
+ \rho_{i,j} \Delta t \varepsilon_{H_{i,j}} \left[ \left( i_{i+1,j}^{n} - 2 i_{i,j}^{n} + i_{i-1,j}^{n} \right) r_{i,j} \Delta r^2 + \left( i_{i,j+1}^{n} - 2 i_{i,j}^{n} + i_{i,j-1}^{n} \right) r_{i,j} \Delta x^2 \right] + 0.5 \left( i_{i+1,j}^{n} - i_{i-1,j}^{n} \right) \Delta x^2 \Delta r
\]

4.5 Non-Condensable Gas Equation Linearization

The non-condensable gas transport equation, in two-dimensions and cylindrical coordinates, is given in Eq. 4.73. The non-condensable gas transport equation is solved in a semi-implicit method. Like the continuity and energy equations the only terms that are new time values, other than in the time derivative, are the velocities in the convection terms.

\[
\frac{\partial \rho \alpha}{\partial t} + \frac{\partial \rho \alpha u}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} \left[ \rho \alpha v r \right] = \rho \varepsilon_{D} \left[ \frac{\partial^2 \alpha}{\partial x^2} + \frac{\partial^2 \alpha}{\partial r^2} + \frac{1}{r} \frac{\partial \alpha}{\partial r} \right] \quad (4.73)
\]

The time derivative term in the above equation is expanded using product rule in Eq. 4.74.
The density is a function of both enthalpy and pressure. The time derivative of the density is further expanded using chain rule in Eq. 4.75

\[
\frac{\partial \rho \alpha}{\partial t} = \rho \frac{\partial \alpha}{\partial t} + \alpha \left[ \frac{\partial \rho}{\partial t} + \frac{\partial \rho}{\partial p} \right] \tag{4.75}
\]

The finite volume formulation of the fully expanded temporal term of the non-condensable gas transport equation is given in Eq. 4.76

\[
\frac{\partial \rho \alpha}{\partial t} = \rho_{i,j} \frac{\alpha_{i,j} - \alpha_{i,j}^n}{\Delta t} + \alpha_{i,j}^n \left[ \frac{\partial \rho}{\partial i} \right]_{i,j} \frac{i_{i,j} - i_{i,j}^n}{\Delta t} + \frac{\partial \rho}{\partial p} \frac{p_{i,j} - p_{i,j}^n}{\Delta t} \] \tag{4.76}

The axial flux term of the noncondensable gas transport equation is given in its finite volume formulation in Eq. 4.77

\[
\frac{\partial \alpha}{\partial x} = \left( \frac{\alpha \rho}{\Delta x} \right)_b u_b - \left( \frac{\alpha \rho}{\Delta x} \right)_r u_r \tag{4.77}
\]

In Eq. 4.77 above, the velocities are taken at the new time at the bottom, signified by subscript \(B\), or top, signified by subscript \(T\), faces of the computational cell. The values with asterisk superscript are donor cell values at the designated face.

The formulation of the radial flux term given in Eq. 4.73 above is expanded in Eq. 4.78

\[
\frac{1}{r} \frac{\partial}{\partial r} \left[ \alpha \rho \nu r \right] = \alpha \rho \frac{\partial \nu}{\partial r} + \rho \nu \frac{\partial \alpha}{\partial r} + \alpha \nu \frac{\partial \rho}{\partial r} + \frac{\alpha \rho \nu}{r} \tag{4.78}
\]
The finite volume formation of the radial term is shown in Eq. 4.79

\[
\frac{1}{r} \frac{\partial}{\partial r} \left[ \alpha \rho v_r \right] = \rho_{i,j} \alpha^n_{i,j} \frac{v_o - v_l}{\Delta r} + \left( \rho_{i,j} \frac{\alpha^n_o - \alpha^n_l}{\Delta r} + \alpha^n_{i,j} \frac{\rho_o - \rho_l}{\Delta r} + \frac{P_{i,j} \alpha^n_{i,j}}{r_{i,j}} \right) v_{i,j} \quad (4.79)
\]

In the above equation, the subscript \( O \) indicates the outside, or \( j + \frac{1}{2} \) face and the subscript \( I \) indicates the inside or \( j - \frac{1}{2} \) face. In this semi-implicit implementation the velocities are new time values while the densities and noncondensible gas terms are old time values.

The final term in the concondensible gas equation is the diffusion term. The finite volume formulation of this term is given in Eq. 4.80

\[
\rho_D \left[ \frac{\partial^2 \alpha}{\partial x^2} + \frac{\partial^2 \alpha}{\partial r^2} + \frac{1}{r} \frac{\partial \alpha}{\partial r} \right] = \rho_{i,j} D_{n+1,j} \left[ \frac{\alpha^n_{i+1,j} - 2\alpha^n_{i,j} + \alpha^n_{i-1,j}}{\Delta x^2} + \frac{\alpha^n_{i,j+1} - 2\alpha^n_{i,j} + \alpha^n_{i,j-1}}{\Delta r^2} + \frac{\alpha^n_{i+1,j} - \alpha^n_{i-1,j}}{2r_{i,j} \Delta r} \right] \quad (4.80)
\]

The gas diffusivity, \( D_c \), is the molecular diffusivity of the noncondensible gas in vapor when the flow is laminar or is equal to the molecular turbulent diffusivity of the flow in the vapor when the flow is turbulent, assuming the Schmidt number is near unity. The density and noncondensible gas concentration are cell centered old time values.
The finite volume formulation of the noncondensable gas transport equation is given in Eq. 4.81

\[ \rho_{i,j} \frac{\alpha_{i,j} - \alpha_{i,j}^{n}}{\Delta t} + \alpha_{i,j}^{n} \left[ \left( \frac{\partial \rho}{\partial i} \right)_{i,j}^{n} i_{i,j} - i_{i,j}^{n} \left( \frac{\partial \rho}{\partial p} \right)_{i,j}^{n} p_{i,j} - p_{i,j}^{n} \right] \]
\[ + \left( \alpha p \right)^{b}_{i,j} u_{B} - \left( \alpha p \right)^{r}_{i,j} u_{r} \]
\[ + \rho_{i,j} \alpha_{i,j}^{n} \frac{v_{o} - v_{f}}{\Delta r} + \left( \rho_{i,j} \frac{\alpha_{o}^{n} - \alpha_{i}^{n}}{\Delta r} + \alpha_{i,j}^{n} \frac{\rho_{o} - \rho_{i}}{\Delta r} + \frac{\rho_{i,n} \alpha_{i,n}^{n}}{r_{i,j}} \right) v_{i,j} \]
\[ = \rho_{i,j} \alpha_{i,j}^{n} \left[ \frac{\alpha_{i,j}^{n} - 2 \alpha_{i,j}^{n} + \alpha_{i,j+1}^{n}}{\Delta x^{2}} + \frac{\alpha_{i,j+1}^{n} - 2 \alpha_{i,j}^{n} + \alpha_{i,j-1}^{n}}{\Delta r^{2}} + \frac{\alpha_{i,j+1}^{n} - \alpha_{i,j-1}^{n}}{2r_{i,j} \Delta r} \right] \]

Eq. (4.81)

The new time axial and radial velocities come from Eq. 4.16 and Eq. 4.33 respectively. These expressions are substituted into Eq. 4.81 resulting in the formulation given in Eq. 4.82.
\[ \frac{\rho_{i,j} \alpha_{i,j}^{*} - \alpha_{i,j}^{e} \rho_{i,j}^{*}}{\Delta t} + \alpha_{i,j}^{n} \left[ \left( \frac{\partial \rho}{\partial t} \right)_{i,j}^{n} i_{i,j}^{n} - i_{i,j}^{n} \right] + \alpha_{i,j}^{n} \left[ \left( \frac{\partial p_{i,j}}{\partial t} \right)_{i,j}^{n} - p_{i,j}^{n} \right] \]

\[ + \left( \frac{\rho \alpha}{\Delta x} \right)_{i,j}^{*} \left[ u_{i+1,j}^{0} + c_{i+1,j}^{0} \left[ p_{i+1,j} - p_{i,j} \right] + c_{i+1,j}^{1} \left[ p_{i+1,j} + p_{i,j} \right] + c_{i+1,j}^{2} \left[ i_{i+1,j} + i_{i,j} \right] \right] - \left( \frac{\rho \alpha}{\Delta x} \right)_{i,j}^{*} \left[ u_{i,j}^{0} + c_{i,j}^{0} \left[ p_{i,j} - p_{i-1,j} \right] + c_{i,j}^{1} \left[ p_{i,j} + p_{i-1,j} \right] + c_{i,j}^{2} \left[ i_{i,j} + i_{i-1,j} \right] \right] \]

\[ + \frac{\rho_{i,j} \alpha_{i,j}^{e}}{\Delta r} \left[ v_{i,j+1}^{0} + d_{i,j+1}^{0} \left[ p_{i,j+1} - p_{i,j} \right] + d_{i,j+1}^{1} \left[ p_{i,j+1} + p_{i,j} \right] + d_{i,j+1}^{2} \left[ i_{i,j+1} + i_{i,j} \right] \right] \]

\[ - \frac{\rho_{i,j} \alpha_{i,j}^{e}}{\Delta r} \left[ v_{i,j}^{0} + d_{i,j}^{0} \left[ p_{i,j} - p_{i,j-1} \right] + d_{i,j}^{1} \left[ p_{i,j} + p_{i,j-1} \right] + d_{i,j}^{2} \left[ i_{i,j} + i_{i,j-1} \right] \right] \]

\[ + \frac{1}{2} \left( \frac{\rho_{i,j} \alpha_{0}^{e} - \alpha_{i,j}^{e} \rho_{i,j}}{\Delta r} + \alpha_{i,j}^{e} \frac{\rho_{i,j} - \rho_{i,j}}{\Delta r} + \frac{\rho_{i,j} \alpha_{i,j}^{e}}{\Delta r} \right) \left[ v_{i,j}^{0} + d_{i,j}^{0} \left[ p_{i,j} - p_{i,j-1} \right] \right] \]

\[ + \frac{1}{2} \left( \frac{\rho_{i,j} \alpha_{0}^{e} - \alpha_{i,j}^{e} \rho_{i,j}}{\Delta r} + \alpha_{i,j}^{e} \frac{\rho_{i,j} - \rho_{i,j}}{\Delta r} + \frac{\rho_{i,j} \alpha_{i,j}^{e}}{\Delta r} \right) \left[ v_{i,j+1}^{0} + d_{i,j+1}^{0} \left[ p_{i,j+1} - p_{i,j} \right] \right] \]

\[ \left[ v_{i,j+1}^{0} + d_{i,j+1}^{0} \left[ p_{i,j+1} - p_{i,j} \right] \right] \]

\[ \left[ v_{i,j+1}^{0} + d_{i,j+1}^{0} \left[ p_{i,j+1} - p_{i,j} \right] \right] \]

\[ = \frac{\rho_{i,j} \epsilon_{D,i,j}}{\Delta x^{2}} \left[ \frac{\alpha_{i,j}^{n} - 2 \alpha_{i,j}^{n} + \alpha_{i,j+1}^{n} - 2 \alpha_{i,j}^{n} + \alpha_{i,j-1}^{n} - \alpha_{i,j}^{n}}{\Delta r^{2}} + \frac{\alpha_{i,j}^{n} - \alpha_{i,j}^{n}}{2 r_{i,j} \Delta r} \right] \]

The implicit terms are kept on the left hand side and the explicit terms are moved to the right hand side in Eq. 4.83.
\[ \rho_{i,j} \frac{\alpha_{i,j}}{\Delta t} + \alpha_{i,j}^{\bar{n}} \left[ \left( \frac{\partial \rho}{\partial i} \right)^{n}_{i,j} + \left( \frac{\partial \rho}{\partial p} \right)^{n}_{i,j} \right] \]

\[ + \frac{\rho \alpha}{\Delta x} \left[ c^{0}_{i,j} \left( p_{i+1,j} - p_{i,j} \right) + c^{1}_{i+1,j} \left( p_{i+1,j} + p_{i,j} \right) + c^{2}_{i+1,j} \left[ i_{i+1,j} + i_{i,j} \right] \right] \]

\[ - \frac{\rho \alpha}{\Delta x} \left[ c^{0}_{i,j} \left( p_{i,j} - p_{i-1,j} \right) + c^{1}_{i,j} \left( p_{i,j} + p_{i-1,j} \right) + c^{2}_{i,j} \left[ i_{i,j} + i_{i-1,j} \right] \right] \]

\[ + \frac{\rho_{i,j} \alpha_{i,j}^{n}}{\Delta r} \left[ d^{0}_{i,j+1} \left( p_{i,j+1} - p_{i,j} \right) + d^{1}_{i,j+1} \left( p_{i,j+1} + p_{i,j} \right) + d^{2}_{i,j+1} \left[ i_{i,j+1} + i_{i,j} \right] \right] \]

\[ - \frac{\rho_{i,j} \alpha_{i,j}^{n}}{\Delta r} \left[ d^{0}_{i,j} \left( p_{i,j} - p_{i,j-1} \right) + d^{1}_{i,j} \left( p_{i,j} + p_{i,j-1} \right) + d^{2}_{i,j} \left[ i_{i,j} + i_{i,j-1} \right] \right] \]

\[ + \frac{1}{2} \left( \rho_{i,j} \frac{\alpha_{i,j}^{n} - \alpha_{i,j}^{n}}{\Delta r} + \alpha_{i,j}^{n} \frac{\rho_{i} - \rho_{i}^{*}}{\Delta r} + \frac{\rho_{i,j} \alpha_{i,j}^{n}}{\Delta r} \right) \left[ d^{0}_{i,j} \left( p_{i,j} - p_{i,j-1} \right) \right] \]

\[ + \frac{1}{2} \left( \rho_{i,j} \frac{\alpha_{i,j}^{n} - \alpha_{i,j}^{n}}{\Delta r} + \alpha_{i,j}^{n} \frac{\rho_{i} - \rho_{i}^{*}}{\Delta r} + \frac{\rho_{i,j} \alpha_{i,j}^{n}}{\Delta r} \right) \left[ d^{0}_{i,j+1} \left( p_{i,j+1} - p_{i,j} \right) \right] \]

\[ = \rho_{i,j} \frac{\alpha_{i,j}^{n}}{\Delta t} + \alpha_{i,j}^{n} \left[ \left( \frac{\partial \rho}{\partial i} \right)^{n}_{i,j} + \left( \frac{\partial \rho}{\partial p} \right)^{n}_{i,j} \right] \]

\[ + u_{i,j}^{0} \left( \frac{\rho \alpha}{\Delta x} \right) - u_{i+1,j}^{0} \left( \frac{\rho \alpha}{\Delta x} \right) + v_{i,j}^{0} \frac{\rho_{i,j} \alpha_{i,j}^{n}}{\Delta r} - v_{i,j+1}^{0} \frac{\rho_{i,j} \alpha_{i,j}^{n}}{\Delta r} \]

\[ - \frac{1}{2} \left( \rho_{i,j} \frac{\alpha_{i,j}^{n} - \alpha_{i,j}^{n}}{\Delta r} + \alpha_{i,j}^{n} \frac{\rho_{i} - \rho_{i}^{*}}{\Delta r} + \frac{\rho_{i,j} \alpha_{i,j}^{n}}{\Delta r} \right) \left( v_{i,j+1}^{0} + v_{i,j+1}^{0} \right) \]

\[ + \rho_{i,j} e_{D_{i,j}} \left[ \frac{\alpha_{i,j}^{n} - 2 \alpha_{i,j}^{n} + \alpha_{i,j}^{n}}{\Delta x^{2}} + \frac{\alpha_{i,j+1}^{n} - 2 \alpha_{i,j+1}^{n} + \alpha_{i,j+1}^{n}}{\Delta r^{2}} + \frac{\alpha_{i,j+1}^{n} - 2 \alpha_{i,j+1}^{n} + \alpha_{i,j+1}^{n}}{2 \Delta r \Delta t} \right] \]
\[ \alpha_{i,j} \left[ \rho_{i,j} r_{i,j} \Delta x^2 \Delta r^2 \right] \]

\[ + p_{i,j} \left[ \alpha_{i,j}^n \left( \frac{\partial \rho}{\partial p}_{i,j} \right)^n r_{i,j} \Delta x^2 \Delta r^2 + \left( c_{i+1,j}^1 - c_{i+1,j}^0 \right) \left( \rho \alpha \right)^{\ast}_{i,j} r_{i,j} \Delta x \Delta r^2 \Delta t \right] \]

\[ - \left( c_{i,j}^0 + c_{i,j}^1 \right) \left( \rho \alpha \right)^{\ast}_{i,j} r_{i,j} \Delta x \Delta r^2 \Delta t \]

\[ + \left( d_{i,j}^{i+1} - d_{i,j}^0 - d_{i,j}^0 - d_{i,j}^1 \right) \rho_{i,j} \alpha_{i,j}^n r_{i,j} \Delta x^2 \Delta r \Delta t \]

\[ + \frac{1}{2} \left( \rho_{i,j} \left( \alpha_{i,j}^0 - \alpha_{i,j}^1 \right) r_{i,j} + \alpha_{i,j}^n \left( \rho_0 - \rho_1 \right) r_{i,j} \right) \left( d_{i,j}^0 + d_{i,j}^1 - d_{i,j}^0 - d_{i,j}^1 \right) \Delta x^2 \Delta r \Delta t \]

\[ \left[ \alpha_{i,j}^n \left( \frac{\partial \rho}{\partial t}_{i,j} \right)^n r_{i,j} \Delta x^2 \Delta r^2 + \left( c_{i+1,j}^1 - c_{i+1,j}^0 \right) \left( \rho \alpha \right)^{\ast}_{i,j} r_{i,j} \Delta x \Delta r^2 \Delta t \right] \]

\[ + i_{i,j} \left[ \rho_{i,j} \alpha_{i,j}^n \left( d_{i,j}^{i+1} - d_{i,j}^0 \right) r_{i,j} \Delta x \Delta r^2 \Delta t \right] \]

\[ + \frac{1}{2} \left( \rho_{i,j} \left( \alpha_{i,j}^0 - \alpha_{i,j}^1 \right) r_{i,j} + \alpha_{i,j}^n \left( \rho_0 - \rho_1 \right) r_{i,j} \right) \left( d_{i,j}^2 + d_{i,j}^1 - d_{i,j}^2 - d_{i,j}^1 \right) \Delta x^2 \Delta r \Delta t \]

\[ + p_{i+1,j} \left[ \left( c_{i+1,j}^0 + c_{i+1,j}^1 \right) \left( \rho \alpha \right)^{\ast}_{i,j} r_{i,j} \Delta x \Delta r^2 \Delta t \right] \]

\[ + i_{i+1,j} \left[ \alpha_{i+1,j}^n \left( \frac{\partial \rho}{\partial t}_{i+1,j} \right)^n r_{i,j} \Delta x^2 \Delta r^2 \Delta t \right] \]

\[ + \frac{1}{2} \left( \rho_{i,j} \left( \alpha_{i,j}^0 - \alpha_{i,j}^1 \right) r_{i,j} + \alpha_{i,j}^n \left( \rho_0 - \rho_1 \right) r_{i,j} \right) \left( d_{i+1,j}^0 + d_{i+1,j}^1 \right) \Delta x^2 \Delta r \Delta t \]

\[ + p_{i+1,j+1} \left[ \left( c_{i+1,j+1}^0 + c_{i+1,j+1}^1 \right) \left( \rho \alpha \right)^{\ast}_{i,j} r_{i,j} \Delta x \Delta r^2 \Delta t \right] \]

\[ + i_{i,j+1} \left[ \rho_{i,j} \alpha_{i,j}^n \left( d_{i,j+1}^2 + d_{i,j+1}^1 \right) r_{i,j} \Delta x^2 \Delta r \Delta t \right] \]

\[ + \frac{1}{2} \left( \rho_{i,j} \left( \alpha_{i,j}^0 - \alpha_{i,j}^1 \right) r_{i,j} + \alpha_{i,j}^n \left( \rho_0 - \rho_1 \right) r_{i,j} \right) \left( d_{i,j}^2 + d_{i,j}^1 \right) \Delta x^2 \Delta r \Delta t \]

\[ + p_{i,j+1} \left[ \left( c_{i,j+1}^0 + c_{i,j+1}^1 \right) \left( \rho \alpha \right)^{\ast}_{i,j} r_{i,j} \Delta x \Delta r^2 \Delta t \right] \]

\[ + \frac{1}{2} \left( \rho_{i,j} \left( \alpha_{i,j}^0 - \alpha_{i,j}^1 \right) r_{i,j} + \alpha_{i,j}^n \left( \rho_0 - \rho_1 \right) r_{i,j} \right) \left( d_{i,j}^1 - d_{i,j}^0 \right) \Delta x^2 \Delta r \Delta t \]

\[ + i_{i,j+1} \left[ \rho_{i,j} \alpha_{i,j}^n \left( d_{i,j+1}^0 - d_{i,j}^0 \right) r_{i,j} \Delta x^2 \Delta r \Delta t \right] \]

\[ + \frac{1}{2} \left( \rho_{i,j} \left( \alpha_{i,j}^0 - \alpha_{i,j}^1 \right) r_{i,j} + \alpha_{i,j}^n \left( \rho_0 - \rho_1 \right) r_{i,j} \right) \left( d_{i,j}^1 - d_{i,j}^0 \right) \Delta x^2 \Delta r \Delta t \]

\[ + \frac{1}{2} \left( \rho_{i,j} \left( \alpha_{i,j}^0 - \alpha_{i,j}^1 \right) r_{i,j} + \alpha_{i,j}^n \left( \rho_0 - \rho_1 \right) r_{i,j} \right) \left( d_{i,j}^1 - d_{i,j}^0 \right) \Delta x^2 \Delta r \Delta t \]
The mass, energy, and noncondensable gas transport equations are solved simultaneously yielding the pressure, enthalpy, and noncondensable gas concentration in each computational cell. The two momentum equations are then solved for the new time velocities at the faces of the continuity cells. The boundary conditions used to solve the equations are discussed in the next section.

4.6 Boundary Conditions

The solution of the flow equations requires several boundary conditions. Several conditions are needed for both the liquid jet and the vapor boundary layer.

Within the liquid jet, two pressure, two enthalpy, two axial momentum, and two radial momentum boundary conditions are needed. At the inside of the jet, at \( r = 0 \), there is a reflective boundary condition. The derivative of the pressure, enthalpy, and axial velocity in the radial direction at the centerline of the jet are zero. In order to implement...
this boundary condition the pressure, enthalpy, and axial velocity in the dummy computational cells at $j-1$ are equal to those in the inner most computational cell. The radial momentum at the inside face of the jet is zero.

At the inlet of the liquid jet the axial velocity at the top face of the jet is set to the inlet velocity. The enthalpy in the dummy computational cell above the liquid jet is fixed to the input enthalpy of the liquid jet. The radial velocity at the top of the jet is zero.

The pressure at the exit of the liquid jet is fixed to the pressure of the problem.

The vapor boundary layer has a similar number of boundary conditions. At the outside of the vapor boundary layer, the enthalpy and pressure in the dummy computational cell are held constant at the problem conditions. The axial velocity is fixed to zero in the dummy cell at the outside of the vapor boundary.

At the top of the vapor boundary layer, the enthalpy in the dummy cell is set to that of the problem.

At the bottom of the boundary layer, the pressure is held constant at the pressure of the problem.

Interface boundary conditions also exist. Both the heat and mass transfer and interfacial shear boundary conditions have been discussed in previous sections of this
chapter. In sections 4.3 and 4.4 the changes to the continuity and energy equations from the implicit heat and mass transfer terms was discussed. Changes also occur to these equations because of the implicit interfacial shear.

As discussed in section 3.4.8 the effects of interfacial shear on the liquid jet are neglected. The changes to the continuity, energy, and noncondensable gas transport equations only take place on the vapor side of the interface. The expression for the axial velocity which is substituted into the three conservation equations, instead of being of the form of Eq. 4.16 is given by Eq. 4.20 which is shown again below.

\[
\begin{align*}
    u_{i,j} &= \left( u_{i,j}^0 + c_{i,j}^0 \left( p_{i,j} - p_{i-1,j} \right) + c_{i,j}^1 \left( p_{i,j} + p_{i-1,j} \right) \right) \left( 1 - K_I \right) \\
    &+ \left( u_{i,j-1}^0 + c_{i,j-1}^0 \left( p_{i,j-1} - p_{i-1,j-1} \right) + c_{i,j-1}^1 \left( p_{i,j-1} + p_{i-1,j-1} \right) \right) K_I \\
    &+ \left( +c_{i,j}^2 \left[ i_{i,j} + i_{i-1,j} \right] \right)
\end{align*}
\]

By applying this set of boundary conditions along with the five equations of motion, a solution to the problem of condensation on a liquid jet can be found. The verification of the above equations of motion and their associated boundary conditions will be discussed in the following section.
4.7 Verification of Stand Alone Liquid Jet Model

The ability of the liquid jet model to properly solve the set of flow equations discussed above must be verified in order to trust the results given by it. A set of several basic test problems were run in order to verify the mass and energy balances in both the vapor boundary layer and liquid jet with and without heat and mass transfer and with and without interfacial shear.

In the case without mass transfer and without interfacial shear, the liquid jet model showed a completely stagnant vapor boundary layer and a uniform liquid jet. This is what would be expected. There is no interaction between the liquid and vapor phases, so the liquid jet flowed vertically downward at constant velocity both radially and axially. The noding for this simulation is shown in Figure 4-4. This figure shows that there are twenty axial continuity cells, meaning there are twenty-one axial momentum cells. It also shows five radial cells in both the liquid jet and vapor boundary layer, which are separated by the extended line.
Figure 4-5 below shows the radial velocity profiles within the liquid jet at selected radial locations for a case with no heat and mass transfer and no interfacial shear. The velocity profile, as expected, is uniform across the jet. Also as expected, the velocity profile is the same at all axial locations within the jet. It therefore follows that mass and energy are conserved within the jet. There is no figure shown of the vapor boundary layer because all of the velocities in both the radial and axial directions are within machine rounding error of zero.
Figure 4-5: Radial Velocity Profile in Liquid Jet with no Heat and Mass Transfer and no Interfacial Shear
When there is interfacial shear, but there is no heat and mass transfer, the velocity profile within the liquid jet is identical to that in Figure 4-5 above. This is because the interfacial shear has been assumed to not affect the liquid jet. The shear force does however accelerate the vapor at the interface. This will induce a radial flow within the jet as well as the axial flow. In order to show that this is correctly solved within the jet a mass balance will be done on the faces of the overall boundary layer. This was performed on a three cell by three cell boundary layer. The noding for this case and the following case is shown in Figure 4-6. The results of the mass balance are shown in Table 4-1.

Figure 4-6: Noding of 3x3 problem
A third case was evaluated with both interfacial shear and heat and mass transfer. Mass and energy balances were performed over both the vapor boundary layer and the liquid jet. This was done on a three cell by three cell boundary layer and a three cell by three cell liquid jet. The results of the mass and energy balances in the vapor boundary are shown in Table 4-2.

### Table 4-1: Mass Balance of Vapor Boundary with No Heat and Mass Transfer

<table>
<thead>
<tr>
<th>Location</th>
<th>Mass Flow Rate (lbm/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass in Top (4)</td>
<td>0.004146</td>
</tr>
<tr>
<td>Mass in Top (5)</td>
<td>0.000329</td>
</tr>
<tr>
<td>Mass in Top (6)</td>
<td>0.000088</td>
</tr>
<tr>
<td>Mass out Bottom (4)</td>
<td>0.005282</td>
</tr>
<tr>
<td>Mass out Bottom (5)</td>
<td>0.000759</td>
</tr>
<tr>
<td>Mass out Bottom (6)</td>
<td>0.0000626</td>
</tr>
<tr>
<td>Mass in Side (1)</td>
<td>0.000672</td>
</tr>
<tr>
<td>Mass in Side (2)</td>
<td>0.000495</td>
</tr>
<tr>
<td>Mass in Side (3)</td>
<td>0.000371</td>
</tr>
<tr>
<td><strong>Total Mass in</strong></td>
<td><strong>0.006103</strong></td>
</tr>
<tr>
<td><strong>Total Mass out</strong></td>
<td><strong>0.006103</strong></td>
</tr>
</tbody>
</table>
As the results in Table 4-2 above show, the liquid jet model conserves both mass and energy within the vapor boundary layer. The results of the mass and energy balance in the liquid jet are given in Table 4-3 below.

Table 4-2: Mass and Energy Balance for Vapor Boundary Layer

<table>
<thead>
<tr>
<th>Location [in/out]</th>
<th>Mass (lbm/s)</th>
<th>Energy (Btu/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top (4) [in]</td>
<td>0.004217</td>
<td>4.8816</td>
</tr>
<tr>
<td>Top (5) [in]</td>
<td>0.000377</td>
<td>0.4362</td>
</tr>
<tr>
<td>Top (6) [in]</td>
<td>0.000289</td>
<td>0.3348</td>
</tr>
<tr>
<td>Bottom (4) [out]</td>
<td>0.006095</td>
<td>7.0561</td>
</tr>
<tr>
<td>Bottom (5) [out]</td>
<td>0.000321</td>
<td>0.3712</td>
</tr>
<tr>
<td>Bottom (6) [out]</td>
<td>-0.000052</td>
<td>-0.0603</td>
</tr>
<tr>
<td>Side (1) [in]</td>
<td>0.002842</td>
<td>3.2907</td>
</tr>
<tr>
<td>Side (2) [in]</td>
<td>0.001599</td>
<td>1.8505</td>
</tr>
<tr>
<td>Side (3) [in]</td>
<td>0.000975</td>
<td>1.1282</td>
</tr>
<tr>
<td>Mass Transfer (1) [out]</td>
<td>0.002475</td>
<td>2.8654</td>
</tr>
<tr>
<td>Mass Transfer (2) [out]</td>
<td>0.000865</td>
<td>1.0001</td>
</tr>
<tr>
<td>Mass Transfer (3) [out]</td>
<td>0.00059</td>
<td>0.6832</td>
</tr>
<tr>
<td><strong>Total In</strong></td>
<td>0.01029</td>
<td>11.92</td>
</tr>
<tr>
<td><strong>Total Out</strong></td>
<td>0.01029</td>
<td>11.92</td>
</tr>
</tbody>
</table>
The results of the mass and energy balances shown in Table 4-3 show that the liquid jet model is able to conserve mass and energy within the liquid jet. The outcomes of the verifications completed above allow us to believe the results of simulations using the liquid jet model.
Chapter 5

Implementation of Jet Model in COBRA-TF

In COBRA-TF, the geometry of the problem being modeled is represented by its hydraulic diameter, volume, cross-sectional area, length, and location of its center of mass. The true shape of the control volumes is not known. COBRA-TF represents the connections between the control volumes as gaps with given resistances, but not the true geometry of the connections. For this reason, the geometry of the problem must be constructed in the liquid jet model and made relatable to the COBRA-TF grid structure. This will allow the flexibility to uniquely model the liquid jet equations and couple them properly to the COBRA-TF mesh. Figure 5-1 shows an example of the jet model sub-grid within a COBRA-TF grid. The solid lines indicate, in two dimensions, the COBRA-TF mesh and the dashed lines represent the proposed sub-grid. The sub-grid exists entirely within a single COBRA-TF channel.
The liquid jet model sub-grid will overlay the COBRA-TF computational mesh. By design, the liquid jet must exist within a single cobra channel. The jet model will use the old timestep values for the steam properties. These values will be used in solving the governing and closure equations for the liquid jet and the surrounding vapor boundary layer. Within the COBRA-TF channel, the interface between the liquid jet subgrid and the COBRA-TF mesh will be simulated by using liquid, vapor, and gas mass, momentum, and energy sources and sinks.

Figure 5-1: Sketch of jet model within COBRA-TF mesh
5.1 COBRA-TF Conservation Equations

The two-fluid, three-field approach used in COBRA-TF contains nine conservation equations [44, 6, 45]. There are four mass equations for the three fields including equations for vapor, continuous liquid, entrained droplets, and noncondensable gases. COBRA-TF is defined as a three-field instead of a four field approach because there are only three vector momentum equations. These are for the vapor, continuous liquid, and entrained droplets. It is assumed that the noncondensable gases travel at the same velocity as the vapor field. This leaves two energy equations, one for vapor and one for liquid. The handling of the energy equations requires that the entrained droplet field and the continuous liquid field be in thermal equilibrium with each other.

The vapor continuity equation is given as

\[
\frac{\partial}{\partial t} \alpha_v \rho_v + \nabla \cdot (\alpha_v \rho_v \vec{U}_v) = \Gamma'' + \nabla \cdot \vec{G}_v^T
\]  

(5-1)

The continuous liquid continuity equation is

\[
\frac{\partial}{\partial t} \alpha_l \rho_l + \nabla \cdot (\alpha_l \rho_l \vec{U}_l) = -\Gamma_l'' - S'' + \nabla \cdot \vec{G}_l^T
\]  

(5-2)

The entrained droplet continuity equation is given as

\[
\frac{\partial}{\partial t} \alpha_e \rho_e + \nabla \cdot (\alpha_e \rho_e \vec{U}_e) = -\Gamma_e'' + S''
\]  

(5-3)

The noncondensable gas continuity equation is given as

\[
\frac{\partial}{\partial t} \alpha_g \rho_g + \nabla \cdot (\alpha_g \rho_g \vec{U}_g) = \Gamma_g'' + \nabla \cdot \vec{G}_g^T
\]  

(5-4)
All four of the continuity equations are written as: Rate change of mass in the volume + Rate of gain by convection = Rate of mass gain by interfacial heat transfer + Rate of entrainment + Rate of mass efflux due to void drift.

The vapor and noncondensable gas energy conservation equation is given as

$$\frac{\partial}{\partial t} \alpha_v \left( \rho_v h_v + \rho_g h_g \right) + \nabla \left( \alpha_v \left[ \rho_v h_v + \rho_g h_g \right] \vec{U}_v \right) = \Gamma'' H_g + q_n + Q_{vv}^m$$

$$- \nabla \left( \alpha_v g_v^T \right) + \alpha_v \frac{\partial P}{\partial t}$$

(5-5)

The energy conservation equation for the continuous liquid and entrained droplets is given as

$$\frac{\partial}{\partial t} (\alpha_l + \alpha_s) \rho_l \dot{h}_l + \nabla (\alpha_l \rho_l \dot{h}_l \vec{U}_l) + \nabla (\alpha_s \rho_s \dot{h}_s \vec{U}_s) = \Gamma'' H_f + q_{jl}$$

$$+ Q_{lv}^m - \nabla (\alpha_l g_{l/l}^T) + (\alpha_l + \alpha_s) \frac{\partial P}{\partial t}$$

(5-6)

The energy conservation equations are of the form: Time rate of change + Rate of gain of energy due to convection = Energy transport from phase change + Interfacial heat transfer + Wall heat flux – Turbulent heat flux + Pressure derivative.

The vapor and noncondensable gas momentum equation is given as

$$\frac{\partial}{\partial t} \alpha_v \left( \rho_v + \rho_g \right) \vec{U}_v + \nabla \left( \alpha_v \left[ \rho_v + \rho_g \right] \vec{U}_v \vec{U}_v \right) = -\alpha_v \nabla P + \alpha_v \left( \rho_v + \rho_g \right) \vec{g}$$

$$+ \tau_{vv}^m - \tau_{ij}^v - \tau_{lv}^v + \left( \Gamma'' \vec{U}_i \right)$$

(5-7)
The continuous liquid momentum equation is

\[
\frac{\partial}{\partial t} \alpha_l \rho_l \vec{U}_l + \nabla \cdot (\alpha_l \rho_l \vec{U}_l \vec{U}_l) = -\alpha_l \nabla P + \alpha_l \rho_l \vec{g} + \tau_{wi}''''
\]

\[
+ \tau_{lv}''''(\Gamma_l''''\vec{U}_l) - (S''''\vec{U}_l)
\]

(5-8)

The momentum equation for the entrained droplet field is given as

\[
\frac{\partial}{\partial t} \alpha_e \rho_e \vec{U}_e + \nabla \cdot (\alpha_e \rho_e \vec{U}_e \vec{U}_e) = -\alpha_e \nabla P + \alpha_e \rho_e \vec{g} + \tau_{we}''''
\]

\[
+ \tau_{ev}''''(\Gamma_e''''\vec{U}_e) + (S''''\vec{U}_e)
\]

(5-9)

The momentum conservation equations are of the form: Rate change of momentum + Rate of change of momentum due to convection = Pressure gradient + Wall shear + Interfacial drag + Momentum exchange due to mass transfer + Momentum exchange from entrainment.

5.2 COBRA-TF Finite Difference Equations

The finite difference equations are written using a combination of old and new time values. If no superscript is used the variable is a new time value. The superscript n represents an old time value and a superscript asterisk indicates an old time value from a donor cell. The continuity equation for the vapor phase becomes

\[
\left[ (\alpha_v \rho_v)_{v} - (\alpha_v \rho_v)^*_{v} \right] \frac{A_{v_{j}}} {\Delta t} = \sum_{KB=1}^{NB} \left[ (\alpha_v \rho_v)^* U_{v_{ij} A_{m_{ij}}} \right]_{KB} - \sum_{KA=1}^{NA} \left[ (\alpha_v \rho_v)^* U_{v_{ij} A_{m_{ij}}} \right]_{KA}
\]

\[
- \sum_{KL=1}^{KK} L_{KL} \left[ (\alpha_v \rho_v)^* W_{i} \right]_{LA} + \frac{\Gamma_{j}} {\Delta X_{j}} + \frac{S_{cv_{j}}}{\Delta X_{j}}
\]

(5-10)
The continuous liquid mass equation is

\[
\frac{\left(\alpha_i \rho_i \right)_j - \left(\alpha_i \rho_i \right)_{j,n}}{\Delta t} A_{c,j} = \sum_{KB=1}^{NB} \frac{\left(\alpha_i \rho_i \right)^* U_{l_j,A_{m_j+1, KB}}}{\Delta X_j}
\]

\[
- \sum_{KA=1}^{NA} \frac{\left(\alpha_i \rho_i \right)^* U_{e_j,A_{m_j+1, KA}}}{\Delta X_j} + \sum_{KL=1}^{NKK} L_{KL} \left[ \left(\alpha_i \rho_i \right)^* W_{L_{K}} \right]_{LA} - \frac{(1-\eta) \Gamma_{j}^*}{\Delta X_j} + \frac{S_j}{\Delta X_j} + \frac{S_{e,j}}{\Delta X_j}
\]

The entrained droplet mass equation is

\[
\frac{\left(\alpha_i \rho_i \right)_j - \left(\alpha_i \rho_i \right)_{j,n}}{\Delta t} A_{c,j} = \sum_{KB=1}^{NB} \frac{\left(\alpha_i \rho_i \right)^* U_{e_j,A_{m_j+1, KB}}}{\Delta X_j}
\]

\[
- \sum_{KA=1}^{NA} \frac{\left(\alpha_i \rho_i \right)^* U_{e_j,A_{m_j+1, KA}}}{\Delta X_j} + \sum_{KL=1}^{NKK} L_{KL} \left[ \left(\alpha_i \rho_i \right)^* W_{L_{K}} \right]_{LA} - \frac{\eta \Gamma_{j}^*}{\Delta X_j} + \frac{S_j}{\Delta X_j} + \frac{S_{e,j}}{\Delta X_j}
\]

In the above set of continuity equations, the axial velocity is given by \( U \) and the transverse velocity is \( W \). The \( \Gamma \) represents vapor generation, \( S \) is the entrainment rate, \( S_c \) is the source term for each phase. The cross-sectional area of the control volume is given by \( A_c \), \( L \) represents the size of the gap between to adjacent volumes, and \( \Delta X \) is the length of the control volume.
The vapor phase momentum equation for the axial direction becomes

\[
\left[\frac{(\alpha_v \left[ \rho_v + \rho_g \right] U_v)_j - (\alpha_v \left[ \rho_v + \rho_g \right] U_v)^n_j}{\Delta t}\right] A_{m_j}
\]

\[
\sum_{KB=1}^{NB} \left[ (\alpha_v \left[ \rho_v + \rho_g \right] U_v)^n_{j_{KB}} \right] A_{m_{KB}} \Delta X_j
\]

\[
-\sum_{KA=1}^{NA} \left[ (\alpha_v \left[ \rho_v + \rho_g \right] U_v)^n_{j_{KA}} \right] A_{m_{KA}} \Delta X_j
\]

\[
+ \sum_{KLB=1}^{NKB} \left[ (\alpha_v \left[ \rho_v + \rho_g \right] U_v)^n_{j_{KLB}} \right] \frac{L_{KLB}}{2}
\]

\[
+ \sum_{KLA=1}^{NKA} \left[ (\alpha_v \left[ \rho_v + \rho_g \right] U_v)^n_{j_{KLA}} \right] \frac{L_{KLA}}{2} - \left( \alpha_v \left[ \rho_v + \rho_g \right] \right)^n_j g A_{m_j}
\]

\[
- \frac{(P_{j+1} - P_j)}{\Delta X_j} \alpha^n_j A_{m_j}
\]

\[-K_w,v \left( 2U_v - U^n_{v,j} \right) - K_{\varepsilon,v} \left[ 2(U_v - U_l)^n_j - (U_v - U_l)^n_j \right]
\]

\[-K_{\varepsilon,v} \left[ 2(U_v - U_e)^n_j - (U_v - U_e)^n_j \right]
\]

\[-\frac{\left[ \Gamma C U_v^n - (1-\eta) \Gamma_{\varepsilon} U_l^n - \eta \Gamma_{\varepsilon} U_e^n \right]}{\Delta X_j} + \frac{S_{n_{vj}}^n}{\Delta X_j} + \tau_{v,j}
\]
The continuous liquid momentum equation is given as

\[
\frac{\left[ (\alpha_1 \rho_1 U_1) - (\alpha_1 \rho_1 U_1)^n \right] A_{m_j}}{\Delta t} = \sum_{KB=1}^{NB} \left[ (\alpha_1 \rho_1 U_1)^* U_{ij}^n \right] A_{mKB} \frac{\Delta X_j}{\Delta X}
\]

\[- \sum_{KA=1}^{NA} \left[ (\alpha_1 \rho_1 U_1) U_{ij,\text{sl}}^n \right] A_{mKA} \frac{\Delta X_j}{\Delta X} + \sum_{KLB=1}^{NKB} \left[ (\alpha_1 \rho_1 U_1) W_{ij}^n \right] \frac{L_{KLB}}{2} \]

\[+ \sum_{KLA=1}^{NKA} \left[ (\alpha_1 \rho_1 U_1) W_{ij}^n \right] \frac{L_{KLA}}{2} - (\alpha_1 \rho_1)^n g A_{m_j} - \frac{(P_{i+1} - P_j)}{\Delta X_j} \alpha^n_i A_{m_j},
\]

\[- K_{w,j} \left( 2U_{ij} - U_{ij}^n \right) - K_{i,vj} \left[ 2(U_v - U_{ij})_j - (U_v - U_{ij})^n \right] \]

\[+ (1 - \eta) \left[ \Gamma_c U_v^n - \Gamma_E U_{ij}^n \right]_j + \frac{S_D U_{ij}^n - S_E U_{ij}^n}{\Delta X_j} + S_{m_j}^n \]

\[+ \tau_{ij}^T = (5-14)\]

The entrained droplet momentum equation is

\[
\frac{\left[ (\alpha_e \rho_e U_e) - (\alpha_e \rho_e U_e)^n \right] A_{m_j}}{\Delta t} = \sum_{KB=1}^{NB} \left[ (\alpha_e \rho_e U_e)^* U_{ej}^n \right] A_{mKB} \frac{\Delta X_j}{\Delta X}
\]

\[- \sum_{KA=1}^{NA} \left[ (\alpha_e \rho_e U_e) U_{ej,\text{sl}}^n \right] A_{mKA} \frac{\Delta X_j}{\Delta X} + \sum_{KLB=1}^{NKB} \left[ (\alpha_e \rho_e U_e) W_{ej}^n \right] \frac{L_{KLB}}{2} \]

\[+ \sum_{KLA=1}^{NKA} \left[ (\alpha_e \rho_e U_e) W_{ej}^n \right] \frac{L_{KLA}}{2} - (\alpha_e \rho_e)^n g A_{m_j} - \frac{(P_{i+1} - P_j)}{\Delta X_j} \alpha^n_e A_{m_j},
\]

\[- K_{w,e} \left( 2U_{ej} - U_{ej}^n \right) - K_{i,vej} \left[ 2(U_v - U_{ej})_j - (U_v - U_{ej})^n \right] \]

\[+ \eta \left[ \Gamma_c U_v^n - \Gamma_E U_{ej}^n \right]_j + \frac{S_D U_{ej}^n - S_E U_{ej}^n}{\Delta X_j} + S_{m_j}^n \]

\[+ \tau_{ej}^T = (5-15)\]
In the momentum equations, $K_w$ and $K_i$ represent wall and interfacial shear coefficients, $\tau^T$ is the fluid-fluid shear, $S_D$ is droplet deposition and $S_E$ is droplet entrainment.

The energy equation for the vapor and noncondensable gas field becomes

$$
\frac{\left[ \left( \alpha_v \left[ \rho_v H_v + \rho_s H_g \right] \right)_j - \left( \alpha_v \left[ \rho_v H_v + \rho_s H_g \right] \right)_n \right] A_{\epsilon_j}}{\Delta t} = 
\sum_{KB=1}^{NB} \left[ \left( \alpha_v \left[ \rho_v H_v + \rho_s H_g \right] \right)^* U_{v_j} A_{m_{j-1}} \right]_{KB} \Delta X_j 
- \sum_{KA=1}^{NA} \left[ \left( \alpha_v \left[ \rho_v H_v + \rho_s H_g \right] \right)^* U_{v_j} A_{m_{j}} \right]_{KA} \Delta X_j 
+ \sum_{KL=1}^{NKK} L_{KL} \left[ \left( \alpha_v \left[ \rho_v H_v + \rho_s H_g \right] \right)^* W_L \right]_{LA} \Delta X_j 
+ \frac{\Gamma_j H_{g,j}}{\Delta X_j} + \frac{q_{v,j}}{\Delta X_j} + \frac{Q_{v,j}^p}{\Delta X_j} - \frac{S_{v,j}}{\Delta X_j} + \frac{Q_{v,j}^T}{\Delta X_j} - \frac{\alpha_{\epsilon_j} (P - P^n) A_{\epsilon_j}}{\Delta t}
$$

(5-16)
The continuous liquid and entrained droplet fields share a single energy equation given as

\[
\left[ \left( (\alpha_l + \alpha_e) \rho_l H_l \right)_j - \left( (\alpha_l + \alpha_e) \rho_l H_l \right)_{j'} \right] A_{ij} = \frac{\Delta t}{\Delta X_j} \sum_{KB=1}^{NR} \left[ (\alpha_l \rho_l H_l)^* U_{ij,A_{m,p+1}} + (\alpha_e \rho_e H_e)^* U_{ei,A_{m,p+1}} \right]
\]

\[
\sum_{KA=1}^{NA} \left[ (\alpha_l \rho_l H_l)^* U_{ij,A_{m,p}} + (\alpha_e \rho_e H_e)^* U_{ei,A_{m,p}} \right]
\]

\[
\sum_{KL=1}^{NKK} L_{KL} \left[ (\alpha_l \rho_l H_l)^* W_{ij,A_{m,p}} + (\alpha_e \rho_e H_e)^* W_{ei,A_{m,p}} \right]_{j'}
\]

\[
\frac{\Gamma H_{ij}}{\Delta X_j} \frac{q_{ij}}{\Delta X_j} + \frac{Q^n_{ij}}{\Delta X_j} + \frac{S_{elij}}{\Delta X_j} + \frac{Q^n_{ij}}{\Delta X_j} + \frac{\alpha_{ij} (P - P^n)}{\Delta t} A_{ij}
\]

In the energy equations, \(q_i\) is the interfacial heat flux, \(Q\) is the heat added from the walls, \(Q^T\) is the turbulent heat exchange.

### 5.3 Changes to Conservation Equation for Liquid Jet Model

In implementing the liquid jet model into COBRA-TF, some modifications will need to be made to the COBRA-TF conservation equations and finite difference formulations. Mass, energy, and momentum sinks need to be added to the vapor and liquid phase equations. The modified vapor equations will be used in control volumes in which the jet model control volumes exist. The modifications to the liquid equations will
be used in the control volume in which the jet ends. An explicit coupling will be used, such that the mass and energy sinks on the vapor and liquid phases will occur solely on the right hand side of the equations.

The sources and sinks that are required for the vapor and liquid phase equations require new variables to be defined. The vapor mass that needs to be removed from the COBRA-TF control volume and into the liquid jet model control volumes is going to be defined as $\Gamma_j$. This mass sink is going to be equal to the mass of vapor that is condensed onto the liquid jet in the liquid jet control volume. The term that will be added to the vapor continuity equation will be $-\frac{\Gamma_j}{\Delta X_j}$. The energy lost will be $-\frac{\Gamma_j H_{8j}}{\Delta X_j}$.

In addition to the mass, energy, and momentum that is removed from the vapor phase, modifications need to be made to the continuous liquid field to account for the liquid jet ending. The mass which will be deposited to the continuous liquid field is defined as $J_l$. The term added to the continuous liquid continuity equation is $+\frac{J_l}{\Delta X_j}$. The term added to the liquid energy equation will be $+\frac{J_l H_{8j}}{\Delta X_j}$. It is assumed that the liquid jet ends into a pool of liquid which is large enough that the momentum addition from the jet is negligible. Therefore there is no modification necessary to the continuous liquid momentum equation. In the energy term, $H_{lj}$ is the enthalpy of the liquid jet.
5.4 Coupling of Jet Model to COBRA-TF

The liquid jet module is coupled to COBRA-TF through an interface module. This module serves as the only link between COBRA-TF and the new liquid jet model. A user indicates the jet model is to be used in the boundary conditions section of the COBRA-TF input deck. A new boundary condition has been created, which calls an initialization routine within the interface module. The new boundary condition requires an additional input line to be read at the end of the boundary condition input card.

This new line contains all of the input information needed by the jet model. The input includes the jet diameter and jet length in feet, the jet inlet temperature in degrees Fahrenheit, the jet inlet velocity in feet per second, the vapor boundary layer diameter in feet, the temperature of the vapor, the number of axial nodes within the jet, the number or radial nodes within the jet, and the number of radial nodes within the boundary layer. The input is read in a single line, with the first six inputs being real numbers with maximum length of ten characters, and the final three inputs are integers with maximum length of five characters. After the jet input line is read, a call is made to the interface module initializing the jet model. Once the jet is initialized the interface module interacts with both the jet model and COBRA-TF in several ways.

The interface is designed such that there are always at least two jet model time steps for each COBRA-TF time step. At the beginning of each COBRA-TF time step, a subroutine is called within the interface module to run the jet model for the time
necessary for the current COBRA-TF time step size. The maximum time step in the liquid jet model is set to be half of the COBRA-TF time step. This ensures that the jet model time step is smaller than the COBRA-TF time step.

At the end of each liquid jet model time step the amount of mass transferred in each axial location is added to a running summation. At the end of the COBRA-TF time step, this running summation is divided by the time step size to give an averaged mass transfer rate over the COBRA-TF time step.

The mass that leaves the bottom of the jet is found in a similar way. At the end of each time step, in each radial node, the product of the axial velocity at the bottom of the jet, the flow area, the liquid density, and the time step size is found. The sums of all of the radial locations are kept in a running summation which at the end of the COBRA-TF time step is divided by the time step size.

During each COBRA-TF time step, right after the linearized interfacial heat transfer subroutine is called in xscheme.f, a subroutine in the interface module is called to add mass and energy terms to the right hand sides of the vapor and liquid continuity and energy conservation equations. This subroutine will remove vapor mass and its associated energy in those COBRA-TF computational mesh cells in which the jet exists. It will also add liquid mass and associated energy into the COBRA-TF computational mesh cell in which the jet ends.
5.5 Verification of Jet Model Coupling to COBRA-TF

The coupling of the liquid jet model to COBRA-TF was verified utilizing a simplified input deck. A COBRA-TF input deck was constructed with no flow boundary conditions at the top and bottom of the vessel and only the liquid jet model as an additional boundary condition. A run was conducted with both the liquid jet and the surrounding vapor at saturation so that there would be no heat or mass transfer. The total mass in the vessel is calculated by COBRA-TF and is output to the graphics file. The change in mass in the vessel with time should be equal to the mass flow rate of the liquid jet, since all of the input mass should also be exiting at the bottom of the jet. In Figure 5-2 on the next page, the total mass in the vessel is shown increasing with time.
Figure 5-2: Total vessel mass versus time
The slope of the regression line best fitting the data is 0.00343. This matches the flow rate of the liquid jet which is 0.0034 lbm/s. This shows that the coupling correctly carries the mass through the jet model and into COBRA-TF. Using the same input deck, but with the jet temperature being below saturation and the jet heat transfer model turned off, such that there is no heat or mass transfer even with varying temperatures, an energy balance was performed. From the energy balance a plot, shown in Figure 5-3, was produced showing the change in energy with time.
Figure 5-3: Energy versus Time
Figure 5-3 shows the change in energy with time from the deposition of mass and energy from the liquid jet model. The slope of the regression of the data is 0.716. This corresponds very closely to the energy rate of the liquid jet which is displayed on the plot as 0.72.

The coupling of the mass transfer mass and energy terms is done in the same fashion as the jet termination terms. Therefore the above verification ensures that those terms are properly connected as well.
Chapter 6

Discussion of Results

With both the stand alone liquid jet model and the coupling of the jet model with COBRA-TF being verified the next step is to validate the models against a set of data. The data set used for validating the liquid jet model was discussed in Chapter 2. The test data from Celata et al contains a wide range of conditions which are listed in Table 2-1. The test contains both turbulent and laminar flow and two different nozzle lengths to investigate the entrance effects. The sections below will present a comparison of the liquid jet model results, both stand alone and coupled to COBRA-TF, to the test data as well as a discussion of several components in the liquid jet model which were varied and evaluated.

6.1 Comparison of Liquid Jet Model to Test Data

The bulk of the discussion will detail the evaluation of the stand alone liquid jet model to predict the jet condensation test data. The stand alone jet model was used to predict both turbulent and laminar jets as well as some jets in the laminar to turbulent transition. The critical Reynolds number for the liquid jet model is believed to be between 2000 and 2500 and is assumed to be 2500 for these calculations. The former is the value given by Lin as discussed in Section 3.4.4. However the free surface of the jet
should allow for the dissipation of energy and possibly delay the onset of turbulent formations. This would lead to a critical Reynolds number slightly higher than that for pipe flow, which is generally accepted to be around 2300. The Transition Reynolds number was determined from the usage of the turbulent eddy diffusivity model. Travis [10] states that the data used to develop his model is valid down to a Reynolds number of 4000. Iasachenko, as is discussed in [16], uses the same data set. Therefore a transition Reynolds number of 4000 is used in the calculations.

The turbulent jet simulation will be discussed first followed by the laminar. The experimental data consisted of an axial temperature profile of the jet from its release into the test vessel to a length of 0.825 inches. As a means to evaluate the performance of the liquid jet model, the average temperature of the jet along its x-axis will be compared to the experimental data.

For all of the test cases which will be discussed below, the same nodalization is used. A sensitivity study was performed on radial and axial node sizes which demonstrated that a converging solution was found with increased numbers of nodes and that an accurate solution was attainable with less than ten nodes axially and less than six nodes radially in both the liquid jet and the vapor boundary layer. From this it was decided to use six radial nodes in both the liquid jet and the vapor boundary layer. However, twenty axial nodes were used in order to obtain a more detailed axial temperature distribution.
6.2 Turbulent Jet Results

A majority of the available test data covers turbulent jets. The ability of the liquid jet model to predict several of these experiments will be shown here. There were 102 total experiments performed. Of these 102, about seventy of them were performed with nozzles the same length as their diameter. For these tests, the flow is assumed to have uniform velocity leaving the nozzle. About fifty of the nearly seventy uniform flow experiments featured turbulent jets. An example radial velocity profile within a liquid jet is shown in Figure 6-1. Several radial enthalpy profiles are shown in Figure 6-2. For both of these figures, test 1.002, which will be discussed below was used.
Figure 6-1: Radial profile of the turbulent axial velocity at the 6 inch location
Figure 6-2: Turbulent enthalpy radial profile at various axial locations
The liquid jet model was able to accurately predict the axial temperature profile of the turbulent liquid jets for most cases. In order to compare the calculated results with the experimental data, the calculated radial jet temperature and velocity distributions were integrated to find the bulk axial temperature profile. Several examples will be shown below. Figure 6-3 shows experiment 1.002, which has a Reynolds number of 5862. This is on the lower end of the turbulent region. This particular experiment consisted of a 0.00328 foot diameter jet (1 mm) at 94.8 F and 13.9 ft/s entering into a saturated steam environment at 240 F.
Figure 6-3: Comparison of Simulation of Run 1.002 to Experimental Data
Figure 6-3 shows good agreement between the liquid jet model and the experimental data. Figure 6-4 shows an experiment with the same diameter jet but with a Reynolds number just over ten thousand.
Figure 6-4: Comparison of Simulation of Run 1.003 to Experimental Data
The agreement shown in Figure 6-4 is not as complete as in the previous plot but is still quite good. The discontinuity that appears at the beginning of the jet in the data may be a result of the use of a mixing cup to measure the jet temperature at locations very close to the jet nozzle. Another experiment with the same 0.00328 ft diameter jet but with a Reynolds number over 25,000 is displayed in Figure 6-5.
Figure 6-5: Comparison of Simulation of Run 1.006 to Experimental Data
The agreement between the simulation and the experimental data shown in Figure 6-5 is again very good. Experiments were also performed on liquid jets with considerably less sub-cooling. Figure 6-6 shows the simulation of a jet with the same 0.00328 ft diameter and a Reynolds number around 7000, but with a jet inlet temperature near 200 F. Another experiment performed with lower sub-cooling on a similarly sized jet with a higher Reynolds number, around 25,000, is shown in Figure 6-7.
Figure 6-6: Comparison of Simulation of Run 1.200 with Experimental Data
Figure 6-7: Comparison of Simulation of Run 1.202 with Experimental Data
Both Figure 6-6 and Figure 6-7 show good agreement between the liquid jet model simulation and the experimental data. In the first plot the model predicts well the entire length of the jet while in the second plot the model slightly under-predicts the jet temperature at the start but matches very well at the end.

The turbulent jet experiments also included some with a larger diameter jet. A 0.00656 ft (2 mm) diameter jet at around 80 F with a Reynolds number around 5000 is shown in Figure 6-8. A similar experiment to that one, with the same diameter and jet temperature, but with a Reynolds number of 12,690 is shown in Figure 6-9. Experiments with the slightly larger jet were also carried out with higher liquid temperatures.

A lesser sub-cooled jet is shown in Figure 6-10, with a Reynolds number of 6103 and a jet temperature of 182 F. Figure 6-11 has another higher temperature jet, 155 F, with a higher Reynolds number of 25,523.
Figure 6-8: Comparison of Simulation of Run 1.013 to Experimental Data
Figure 6-9: Comparison of Simulation of Run 1.016 to Experimental Data
Figure 6-10: Comparison of Simulation of Run 1.211 to Experimental Data
Figure 6-11: Comparison of Simulation of Run 1.215 to Experimental Data
In all four of the plots shown above, with 0.00656 ft diameter jets, the simulations display good agreement with the experimental data. The lower Reynolds number jets, which are in Figure 6-8 and Figure 6-10 both slightly over predict the jet temperature over the entire length. The higher Reynolds number jet with higher sub-cooling, shown in Figure 6-9, demonstrates good agreement at the start of the jet, but over predicts by the end. The data at the bottom of the jet appears to asymptote as if it were reaching saturation, which it should not be doing for another seventy degrees, and this contributes to the over prediction by the liquid jet model. The flatter temperature profile shown in the data could be a result of depressurization in the test vessel from condensing more steam than the boiler can replace. The simulation in Figure 6-11 shows good agreement with the data along the entire length of the jet.

The above plots have demonstrated that the liquid jet model is able to simulate condensation on a turbulent liquid jet with confidence. The results here are indicative of most of the experiments featuring turbulent jets.

6.3 Turbulence modeling

Several turbulence models were investigating in the course of this work. The three models that were explored were by Isachenko[16], Hinze[22], and Travis[10]. Both the models by Isachenko and Hinze are given for turbulence within jets, however the model by Isachenko is based off of data by Nikuradze, which is pipe flow data. The
turbulence model by Travis is also derived from pipe flow data by Nikuradze. Travis’s model is given in Eq. 6.1.

\[
\frac{\varepsilon}{\nu} = 0.0028 \text{Re}^{-0.933}
\]  \hspace{1cm} (6.1)

This is very similar in form to both the model by Isachenko, given in Eq. 6.2

\[
\frac{\varepsilon}{\nu} = 0.00025 \text{Re}
\]  \hspace{1cm} (6.2)

And to the model by Hinze in Eq. 6.3

\[
\frac{\varepsilon}{\nu} = 0.13 \text{Re}
\]  \hspace{1cm} (6.3)

All three of the models depend on the Reynolds number and the exponent to which Travis’ model is raised is very close to unity. The coefficients on all three differ greatly however.

The model by Hinze has the largest coefficient, most likely because his model is based on air jets flowing in air. A jet flowing in a similar density fluid would tend to mix and spread much more than a more dense jet flowing in a less dense fluid. For this reason Hinze’s turbulence model was eventually rejected.

The other two turbulence models’ coefficients differ by an order of magnitude. A plot of the two eddy diffusivity coefficients over the range of Reynolds numbers of interest is shown in Figure 6-12.
Figure 6-12: Isachenko and Travis Models for Eddy Diffusivity Ratio
From Figure 6-12 it can be observed that there is a large difference between the two turbulence models, especially at higher Reynolds numbers. During the development of the liquid jet model, the Hinze and Travis models were implemented first. Using either of these turbulence models lead to a large amount of instability and severe over prediction of the jet temperature. The model developed by Isachenko was then implemented in the liquid jet model. This model, which is currently used, ran more stably and as shown in [16] agrees well with the experimental data.

6.4 Laminar Jet Results

The liquid jet model also has the capability to simulate laminar jets and in some cases is able to match with experimental data quite well. An example laminar jet axial velocity radial profile is shown in Figure 6-13. Several laminar enthalpy radial profiles are shown in Figure 6-14. The jet shown in these profiles is from test 1.220, which will be discussed below.
Figure 6-13: Radial profile of the laminar jet axial velocity at the 6 inch location
Figure 6-14: Laminar jet enthalpy radial profile at several axial locations
An example of this is shown in Figure 6-15. However, it is much more common that the liquid jet model would tend to under predict the liquid jet temperature as shown in Figure 6-16. Both of these are larger diameter jets, 0.0164 ft (5 mm), with lesser superheat, inlet temperature around 170 F. An experiment with more liquid jet subcooling is shown in Figure 6-17 and an experiment featuring a 0.00656 ft (2 mm) diameter jet is shown in Figure 6-18.
Figure 6-15: Comparison of Simulation of Run 1.220 with Experimental Data
Figure 6-16: Comparison of Simulation of Run 1.221 with Experimental Data
The first plot above, Figure 6-15, shows that the liquid jet model agrees well with some of the experimental data and the fourth plot Figure 6-18 demonstrates that for other tests the model is agrees well with the temperature at the bottom of the jet. The middle two plots, Figure 6-16 and Figure 6-17, show that the jet also has a tendency to under predict the jet temperature along the entire length of the jet for laminar jets.

The reason for this might be an under prediction of the amount of mixing taking place within the laminar jet. The deposition of new mass from mass transfer might impact the mixing more than the code predicts. Also surface effects could play a large role. Surface wavelets or jet breakup would cause changes in both the heat transfer area and increase the amount of mixing within the jet. Another option explored was the choice of the laminar heat transfer coefficient.

Different laminar heat transfer coefficients were tried in the liquid jet model. The first was using a constant Nusselt number but accounting for entrance effects using an exponential decay similar to convective enhancement in spacer grid modeling. The constant value of the Nusselt number was 4.364 and the entrance effects were modeled using Eq. 3.27. The second model, by Isachenko is given in Eq. 3.28 was developed to model condensation on water jets in air. The results from both models in predicting the temperature rise along the length of the jet were identical. This proves that the heat transfer coefficient is most likely not the driving force in the inaccuracy of liquid jet model to simulate laminar jets. The ability of the liquid jet model to predict the temperature rise in transition Reynolds number jets also points to this.
Figure 6-17: Comparison of Simulation of Run 1.021 with Experimental Data
Figure 6-18: Comparison of Simulation of Run 1.011 with Experimental Data
6.5 Transition Jet Results

The transition Reynolds number jets were well predicted using the liquid jet model. The heat transfer coefficient at transition Reynolds numbers was found by using a linear ramp between the laminar heat transfer coefficient and the turbulent heat transfer coefficient. The ability of the liquid jet model to simulate these conditions is shown in Figure 6-19 and Figure 6-20. Figure 6-19 shows experiment 1.121 which as a Reynolds number of 3076, a jet diameter of 0.00656 ft (2 mm), and a jet inlet temperature of 99 F. The second plot, Figure 6-20, shows experiment 1.110. This test has a Reynolds number of 3475, the same diameter as in the previous plot, but a higher temperature of 143 F.

As demonstrated in both of these plots, the liquid jet models shows good agreement with the experimental data for transition Reynolds number jets. The turbulent mixing that is present even in these lower Reynolds number flows indicates that mixing is important even at these lower velocity flows, perhaps in laminar flows as well, and supports the theory proposed in the previous section.
Figure 6-19: Comparison of Simulation of Run 1.121 with Experimental Data
Figure 6-20: Comparison of Simulation of Run 1.110 with Experimental Data
6.6 Validation of Liquid Jet Model Coupling with COBRA-TF

In order to validate that the liquid jet model is functioning properly when coupled with COBRA-TF several experiments were simulated with the coupled code. The expectation for a properly functioning coupled code would be a predicted jet axial temperature profile which is similar to that predicted using the stand alone liquid jet model and also one that agrees well with the experimental data.

In order to validate the liquid jet version of COBRA-TF an input deck was developed to model the test vessel used in the Celata et al experiment. The input deck featured a single channel with several axial nodes. The computational cell at the bottom of the vessel was made artificially wide to allow any water to build up in a liquid layer there rather than to fit a flow or pressure boundary condition. No flow boundary conditions were placed at both the top and bottom of the test vessel so that no flow can exit or enter at the ends of the vessel. A steam source was added towards the top of the vessel. This is simulated using a pressure source boundary condition which should act to maintain the pressure in the vessel by adding the mass of vapor needed. The liquid jet model is called to simulate a jet beginning towards the top of the test vessel.

In order to show that liquid jet COBRA-TF is able to predict the jet condensation experiments as well as the stand alone liquid jet model the results of simulations of an experiment using both codes as well as the test data are plotted below. Figure 6-21 shows experiment 1.002, which is also displayed in 6.2. This test features a 0.00328 ft
diameter jet entering the test vessel at 13.48 ft/s with a temperature of 94 F. The
Reynolds number of the jet is 5862. Figure 6-21 clearly shows good agreement between
the stand alone liquid jet model and liquid jet COBRA-TF. The two predictions differ
slightly in their estimation of the jet temperature towards the top of the jet but are almost
identical for most of this length. The plot also shows that liquid jet COBRA-TF
prediction shows good agreement with the jet axial temperature profile given in the
experimental data.
Figure 6-21: Comparison of Simulation of Run 1.002 using both Stand Alone and Coupled Liquid Jet Models
A similar plot to Figure 6-21 above, with the addition of a COBRA-TF simulation of the jet using a mass flow boundary condition is shown in Figure 6-22. To create this plot, the input deck used in the coupled case was modified. The jet boundary condition was replaced with a mass source boundary condition at the same location. The liquid in this simulation was treated as a liquid film. The results in Figure 6-22 show a clear improvement from COBRA-TF in the coupled liquid jet COBRA-TF.
Figure 6-22: Comparison of Simulation of Run 1.002 using Stand Alone and Coupled Liquid Jet Models and COBRA-TF
Chapter 7

Conclusions and Recommendations for Future Work

Condensation heat transfer is an important phenomenon in nuclear reactor safety systems and safety calculations. The increasing use of condensation heat transfer, including condensation on jets of water, in safety systems puts added pressure to correctly model this phenomenon with thermal-hydraulic system and sub-channel analysis codes.

The problem of condensation on liquid jets has been studied for some time. Many of the pertinent works are discussed in Chapter 2. Several experiments have been performed exploring condensation on water jets both with and without noncondensable gases present. A data set was found which was well suited for validating a liquid jet model. The Celata et al experiment was chosen because it included in its data set axial temperature profiles along the length of the water jets.

The goal of this work was to develop a stand alone module with which to simulate condensation on a liquid jet and then implement it within a sub-channel analysis code to improve that codes handling of jet condensation. The model created solves two-dimensional heat and flow equations in both a liquid jet, the vapor boundary layer surrounding the jet, and the interactions between the jet and the vapor. The formulations used to solve this problem and the relationships needed to close the equations are
discussed in Chapter 3. The linearization of the governing equations as well as the boundary conditions used to solve the set of equations is discussed in Chapter 4. The coupling of the model to COBRA-TF was discussed in Chapter 5.

The stand alone liquid jet model was verified to conserve both energy and mass in Chapter 4. Verification that the coupling conserves mass and energy followed in Chapter 5. The results of the stand alone liquid jet model and the coupled liquid jet COBRA-TF were discussed in Chapter 6. There it was shown that the liquid jet model vastly improves the ability of COBRA-TF to model condensation on liquid jets.

The liquid jet model has been demonstrated to accurately simulate the heat and mass transfer on turbulent jets. The jet model does however display some instability when presented with high velocity flows. In general these instabilities do not impede the codes ability to accurately predict the temperatures in the jet. An attempt has been made to classify the instability and it appears at high Reynolds number flow when mass transfer is occurring. The instability shows as pressure oscillations in the liquid terms which can lead to variations in the jet velocities. The instability may be caused by neglecting the momentum transfer during mass transfer. This leads to the first recommendation for future work.

It might be advisable to add the axial momentum transfer due to mass transfer into the axial momentum equation. It is currently assumed that the mass is added from the vapor into the liquid with no associated axial momentum. This is made more difficult
because of the method used to solve the momentum equations. A possible solution would be to solve for the new time velocities simultaneously with the new time pressures, enthalpies, and noncondensable gas concentrations. Such a solution scheme would be more computationally taxing but would offer a simpler implementation.

The prediction of laminar jets is another area which could be improved with additional study. The current liquid jet model tends to under predict the temperature within the jet, most likely due to an under prediction of the eddy diffusivity and mixing, as shown in Chapter 6. Additional experimental studies examining the transition from laminar to turbulent flow could be helpful in finding more accurate solutions to these problems.

On a related note to the treatment of the laminar to turbulent transition are the effects of surface variation both in laminar and turbulent jets. Accurately characterizing the fluctuations at the surface might lead to more accurate predictions of radial flow, eddy diffusivities of momentum and heat, and mixing within the jet at both laminar and turbulent Reynolds numbers. Having an idea of the surface shape would also allow more accurate calculations of the surface area, which would give better estimations of the interfacial shear and heat and mass transfer terms.

Following a more accurate determination of the jet surface, a further area for improvement in the liquid jet model would be to develop and implement a model for jet breakup. Efforts made in this area would be of benefit to predicting the heat and mass
transfer as well as the interfacial shear in the liquid jet model. A more physical computational model will lead to more accurate solutions.

Another area for improvement would be to allow the vapor boundary layer to be more general and account for superheat within the vapor. This would not only make the code more general and more widely applicable but would make modeling condensation with noncondensables more accurate. When noncondensable gases are present, there is a tendency for the local vapor partial pressure near the interface to decrease. This lowers the dew point and creates superheated vapor at the interface even if the bulk fluid is not superheated. Implementing handling of superheated vapor within the stand alone liquid jet model would be a relatively trivial exercise. Most of the necessary foundation is already in place. However doing so will necessitate the next area marked for improvement.

Allowing for superheated vapor in the stand alone liquid jet model would most likely require implicit coupling with COBRA-TF. Variation in the vapor temperature or enthalpy in the coupled liquid jet model would necessitate knowledge of the new time enthalpy in the vapor of the COBRA-TF computation.

The final suggestion for future work is in the area of condensation in the presence of noncondensable gases. This recommendation would most likely require a large amount of new research but would be extremely beneficial in extending the usefulness of the liquid jet model. Experimental data is needed to validate and improve the handling
on noncondensable gases in the liquid jet model. Effort developing a more physical model for the effects on noncondensable gases on condensation would also be of great benefit.
Bibliography


Appendix A

Experimental Data from Celata et al Test

DATA FORMAT:

Run number
h/d   d[mm]
Tsteam[^C]  Tjet,in[^C]  flow rate [l/min]
N = number of experimental data
quota 1^ point    temperature 1^ point
quota 2^ point    temperature 2^ point
"    "
"    "
quota N^ point    temperature N^ point

START OF DATA

<table>
<thead>
<tr>
<th>Run number</th>
<th>h/d</th>
<th>d[mm]</th>
<th>Tsteam[^C]</th>
<th>Tjet,in[^C]</th>
<th>flow rate [l/min]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.001</td>
<td>1</td>
<td>1</td>
<td>117.0714</td>
<td>34.7</td>
<td>.1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>.01</td>
<td>70.05393</td>
<td>74.84416</td>
<td>88.02225</td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>.05</td>
<td>100.0581</td>
<td>104.8973</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>.15</td>
<td>108.5378</td>
<td>110.7268</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Column 1</td>
<td>Column 2</td>
<td>Column 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.002</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>115.8489</td>
<td>34.89863</td>
<td>.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.0001</td>
<td>34.89863</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>45.93054</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>49.57881</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>71.25186</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>82.02818</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>94.03034</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>100.0581</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>104.8973</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>107.3232</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.003</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>114.6275</td>
<td>31.19114</td>
<td>.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>31.19114</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>38.59103</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>43.49061</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.015</td>
<td>48.36422</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>73.6469</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>78.43568</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>84.42454</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>92.82731</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>97.162</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>97.64432</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>97.64432</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.004</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>122.8741</td>
<td>28.71144</td>
<td>.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>28.71144</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>37.06669</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>45.44306</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.015</td>
<td>54.42329</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>66.45775</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>82.02818</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>85.62329</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>101.2664</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>110.9703</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>112.6757</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>114.6275</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.006</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>112.1882</td>
<td>29.456</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>29.456</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>45.93054</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>61.65458</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>52.24416</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>73.6469</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>81.54906</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>92.34631</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>94.03034</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>97.64432</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.011</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>118.295</td>
<td>42.26821</td>
<td>.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>42.26821</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>56.83796</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>65.25798</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>79.15405</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>91.62503</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>97.64432</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>100.5413</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>102.4757</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.012</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>113.4073</td>
<td>34.89863</td>
<td>.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>34.89863</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>47.87797</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>66.9375</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>80.83047</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>87.54228</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>89.22255</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>94.03034</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.013</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>114.6275</td>
<td>26.22575</td>
<td>.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>26.22575</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>45.93054</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>59.24822</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>72.4495</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>80.83047</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>84.66426</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>87.54228</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.014</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>115.8489</td>
<td>24.98081</td>
<td>.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>24.98081</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>32.67591</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>48.36422</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>58.04360</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>9.999999E-02</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>68.85565</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>76.04133</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>78.43568</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.015</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>116.5823</td>
<td>23.7346</td>
<td>.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>23.7346</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>29.95207</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>45.93054</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>58.04360</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>9.999999E-02</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>68.85565</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>76.04133</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>77.95680</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.016</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>117.0714</td>
<td>27.96634</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>27.96634</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>32.42861</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>57.65695</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>67.65646</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>9.999999E-02</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>72.44950</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>80.35146</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>80.83047</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.017</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>109.9968</td>
<td>22.48719</td>
<td>1.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>22.48719</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>28.71144</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>32.42861</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>60.53781</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>9.999999E-02</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>72.52019</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>78.02818</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>79.22618</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.02</td>
<td>1</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>-----</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>117.0714</td>
<td>39.81844</td>
<td>.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>39.81844</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>54.42329</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>60.45185</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>68.85565</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>89.94302</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>95.71591</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>98.12677</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>100.0581</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.021</th>
<th>1</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>117.0714</td>
<td>31.93382</td>
<td>.2</td>
</tr>
<tr>
<td>0</td>
<td>31.93382</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>45.19923</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>48.36422</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>56.83796</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>71.25186</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>82.02818</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>88.02225</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>92.34631</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.022</th>
<th>1</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>114.6275</td>
<td>24.98081</td>
<td>.4</td>
</tr>
<tr>
<td>0</td>
<td>24.98081</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>28.71144</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>34.89863</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>42.26821</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>50.79196</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>60.45185</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>70.05393</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>74.84416</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.023</th>
<th>1</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>114.6275</td>
<td>27.22071</td>
<td>.6</td>
</tr>
<tr>
<td>0</td>
<td>27.22071</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>28.71144</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>31.93382</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>37.85377</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>45.93054</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>56.35539</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>64.05756</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>68.85565</td>
<td></td>
</tr>
<tr>
<td>Column</td>
<td>1.024</td>
<td>106.275</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td>0</td>
<td>23.98394</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>27.46931</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>30.44789</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>37.36192</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>44.22326</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>61.65458</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>65.25798</td>
<td></td>
</tr>
<tr>
<td>1.025</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>112.6757</td>
<td>22.48719</td>
</tr>
<tr>
<td>0</td>
<td>22.48719</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>26.22575</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>28.71144</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>34.65193</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>41.04416</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>50.30687</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>57.32035</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>59.24822</td>
<td></td>
</tr>
<tr>
<td>1.026</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>110.9703</td>
<td>22.48719</td>
</tr>
<tr>
<td>0</td>
<td>22.48719</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>24.48247</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>25.60344</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>30.44789</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>36.99286</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>43.77881</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>49.57881</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>52.00372</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>161.7154</td>
<td>73.6469</td>
</tr>
<tr>
<td>0</td>
<td>73.6469</td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>90.42346</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>97.64432</td>
<td></td>
</tr>
<tr>
<td>.015</td>
<td>102.4757</td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>109.7535</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>118.295</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>133.0653</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>141.7475</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>146.7264</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>151.7155</td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>Column 1</td>
<td>Column 2</td>
</tr>
<tr>
<td>--------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td>1.101</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>161.7154</td>
<td>52.00372</td>
<td>.2</td>
</tr>
<tr>
<td>10</td>
<td>52.00372</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>52.00372</td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>56.83796</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>54.42329</td>
<td></td>
</tr>
<tr>
<td>.015</td>
<td>92.82731</td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>108.7809</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>115.8489</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>134.303</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>144.2356</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>146.9756</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>151.7155</td>
<td></td>
</tr>
<tr>
<td>1.102</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>160.4643</td>
<td>29.95207</td>
<td>.4</td>
</tr>
<tr>
<td>10</td>
<td>29.95207</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>29.95207</td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>49.57881</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>100.0581</td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>112.1882</td>
<td></td>
</tr>
<tr>
<td>.04</td>
<td>118.295</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>121.9728</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>140.5047</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>146.7264</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>149.2198</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>151.7155</td>
<td></td>
</tr>
<tr>
<td>1.103</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>159.2135</td>
<td>29.95207</td>
<td>.53</td>
</tr>
<tr>
<td>10</td>
<td>29.95207</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>29.95207</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>49.57881</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>92.82731</td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>109.7535</td>
<td></td>
</tr>
<tr>
<td>.04</td>
<td>121.9728</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>131.8287</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>145.4806</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>151.2162</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>154.2132</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>154.713</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
<td>-------</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>160.9647</td>
<td>61.89502</td>
<td>.15</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>61.89502</td>
</tr>
<tr>
<td>0.01</td>
<td>76.04133</td>
<td></td>
</tr>
<tr>
<td>0.02</td>
<td>83.22618</td>
<td></td>
</tr>
<tr>
<td>0.03</td>
<td>86.82251</td>
<td></td>
</tr>
<tr>
<td>0.04</td>
<td>91.62503</td>
<td></td>
</tr>
<tr>
<td>0.05</td>
<td>102.4757</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>121.9728</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>131.0872</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>134.303</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>140.5047</td>
<td></td>
</tr>
<tr>
<td>1.11</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>161.7154</td>
<td>59.24822</td>
<td>.2</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>59.24822</td>
</tr>
<tr>
<td>0.005</td>
<td>65.25798</td>
<td></td>
</tr>
<tr>
<td>0.01</td>
<td>70.05393</td>
<td></td>
</tr>
<tr>
<td>0.02</td>
<td>77.23849</td>
<td></td>
</tr>
<tr>
<td>0.03</td>
<td>80.83047</td>
<td></td>
</tr>
<tr>
<td>0.04</td>
<td>86.82251</td>
<td></td>
</tr>
<tr>
<td>0.05</td>
<td>92.58679</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>114.6275</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>122.2184</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>128.1248</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>132.5705</td>
<td></td>
</tr>
<tr>
<td>1.112</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>161.7154</td>
<td>34.89863</td>
<td>.4</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>34.89863</td>
</tr>
<tr>
<td>0.005</td>
<td>38.59103</td>
<td></td>
</tr>
<tr>
<td>0.01</td>
<td>42.26821</td>
<td></td>
</tr>
<tr>
<td>0.02</td>
<td>61.65458</td>
<td></td>
</tr>
<tr>
<td>0.03</td>
<td>68.85565</td>
<td></td>
</tr>
<tr>
<td>0.05</td>
<td>76.04133</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>92.82731</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>106.1097</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>114.6275</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>121.9728</td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td>Temperature</td>
<td>Vibration</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td>-----------</td>
</tr>
<tr>
<td>0</td>
<td>27.46931</td>
<td>1.113</td>
</tr>
<tr>
<td>.005</td>
<td>29.95207</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>32.92315</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>54.42329</td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>64.05756</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>73.6469</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>90.42346</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>102.9597</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>112.1882</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>119.5198</td>
<td></td>
</tr>
<tr>
<td>1.114</td>
<td>27.46931</td>
<td>1.113</td>
</tr>
<tr>
<td>1</td>
<td>27.46931</td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>29.95207</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>32.92315</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>54.42329</td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>64.05756</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>73.6469</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>90.42346</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>102.9597</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>112.1882</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>119.5198</td>
<td></td>
</tr>
<tr>
<td>1.115</td>
<td>27.46931</td>
<td>1.113</td>
</tr>
<tr>
<td>9</td>
<td>27.46931</td>
<td></td>
</tr>
<tr>
<td>.005</td>
<td>29.95207</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>31.93382</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>52.24591</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>80.83047</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>101.2664</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>112.1882</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>120.7458</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>125.6607</td>
<td></td>
</tr>
<tr>
<td>1.116</td>
<td>27.46931</td>
<td>1.113</td>
</tr>
<tr>
<td>8</td>
<td>27.46931</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>37.36192</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>49.57881</td>
<td></td>
</tr>
<tr>
<td>.03</td>
<td>69.43881</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>96.4073</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>99.9728</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>119.593</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>120.0653</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>1.12</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>.01</td>
<td>56.83796</td>
</tr>
<tr>
<td></td>
<td>.15</td>
<td>107.3232</td>
</tr>
<tr>
<td>1.121</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>.01</td>
<td>37.36192</td>
</tr>
<tr>
<td></td>
<td>.15</td>
<td>90.42346</td>
</tr>
<tr>
<td>1.122</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>.01</td>
<td>27.46931</td>
</tr>
<tr>
<td></td>
<td>.15</td>
<td>85.62329</td>
</tr>
<tr>
<td>1.123</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>.01</td>
<td>25.2299</td>
</tr>
<tr>
<td></td>
<td>.15</td>
<td>85.62329</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td>1.124</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>24.98081</td>
</tr>
<tr>
<td></td>
<td>.05</td>
<td>9.999999E-02</td>
</tr>
<tr>
<td></td>
<td>.2</td>
<td>91.62503</td>
</tr>
<tr>
<td></td>
<td>1.125</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>24.2083</td>
</tr>
<tr>
<td></td>
<td>.05</td>
<td>60.45185</td>
</tr>
<tr>
<td></td>
<td>.2</td>
<td>94.03034</td>
</tr>
<tr>
<td></td>
<td>1.126</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>24.23323</td>
</tr>
<tr>
<td></td>
<td>.05</td>
<td>66.45775</td>
</tr>
<tr>
<td></td>
<td>.2</td>
<td>97.64432</td>
</tr>
<tr>
<td></td>
<td>1.21</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>83.22618</td>
</tr>
<tr>
<td></td>
<td>.05</td>
<td>107.3232</td>
</tr>
<tr>
<td></td>
<td>.2</td>
<td>130.0991</td>
</tr>
<tr>
<td>1.211</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>146.7264</td>
<td>83.22618</td>
<td>.2</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>83.22618</td>
</tr>
<tr>
<td>.01</td>
<td>88.02225</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>96.67983</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>106.1097</td>
<td></td>
</tr>
<tr>
<td>9.999999E−02</td>
<td>117.0714</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>124.6763</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>129.3584</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>133.0653</td>
<td></td>
</tr>
<tr>
<td>1.212</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>145.4806</td>
<td>74.84416</td>
<td>.4</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>74.84416</td>
</tr>
<tr>
<td>.01</td>
<td>78.43568</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>90.42346</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>99.33356</td>
<td></td>
</tr>
<tr>
<td>9.999999E−02</td>
<td>108.5378</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>114.6275</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>120.5005</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>124.4303</td>
<td></td>
</tr>
<tr>
<td>1.213</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>146.7264</td>
<td>71.25186</td>
<td>.6</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>71.25186</td>
</tr>
<tr>
<td>.01</td>
<td>76.04133</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>89.94302</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>98.85073</td>
<td></td>
</tr>
<tr>
<td>9.999999E−02</td>
<td>107.3232</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>114.6275</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>119.5198</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>124.4303</td>
<td></td>
</tr>
<tr>
<td>1.214</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>144.2356</td>
<td>70.05393</td>
<td>.8</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>70.05393</td>
</tr>
<tr>
<td>.01</td>
<td>73.6469</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>90.42346</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>102.4757</td>
<td></td>
</tr>
<tr>
<td>9.999999E−02</td>
<td>109.7535</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>115.1159</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>119.5198</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>124.4303</td>
<td></td>
</tr>
<tr>
<td>1.215</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.01</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.999999E-02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td></td>
<td>1.216</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.01</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.999999E-02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td></td>
<td>1.217</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.01</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.999999E-02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td></td>
<td>1.22</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.999999E-02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td>Value</td>
<td>0</td>
<td>.01</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>8</td>
<td>76.04133</td>
<td>84.42454</td>
</tr>
<tr>
<td>1.222</td>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>73.6469</td>
<td>74.84416</td>
</tr>
<tr>
<td>1.223</td>
<td>1.224</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>68.85565</td>
<td>71.25186</td>
</tr>
<tr>
<td>1.224</td>
<td>1.224</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>68.85565</td>
<td>70.77273</td>
</tr>
<tr>
<td>1.225</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>142.9912</td>
<td>68.13648</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>68.13648</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>69.33501</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>72.20998</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>76.04133</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>80.83047</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>85.14374</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>89.94302</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>92.82731</td>
<td></td>
</tr>
<tr>
<td>1.226</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>145.4806</td>
<td>66.9375</td>
<td>1.2</td>
</tr>
<tr>
<td>0</td>
<td>66.9375</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>68.85565</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>71.25186</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>75.44275</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>80.83047</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>85.62329</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>90.42346</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>94.03034</td>
<td></td>
</tr>
<tr>
<td>1.227</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>143.6133</td>
<td>65.85793</td>
<td>1.4</td>
</tr>
<tr>
<td>0</td>
<td>65.85793</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>67.89673</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>70.05393</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>74.84416</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>80.83047</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>85.62329</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>89.22255</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>92.82731</td>
<td></td>
</tr>
<tr>
<td>1.13</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>145.4806</td>
<td>38.59103</td>
<td>.1</td>
</tr>
<tr>
<td>0</td>
<td>38.59103</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>56.83796</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>61.65458</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>77.23849</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>107.3232</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>113.4073</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>117.0714</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>124.4303</td>
<td></td>
</tr>
<tr>
<td>1.131</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>145.4806</td>
<td>36.13112</td>
<td>.2</td>
</tr>
<tr>
<td>0</td>
<td>36.13112</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>52.00372</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>54.42329</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>62.85646</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>73.6469</td>
<td></td>
</tr>
<tr>
<td>.1</td>
<td>90.42346</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>100.0581</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>109.7535</td>
<td></td>
</tr>
<tr>
<td>1.132</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>145.9789</td>
<td>24.98081</td>
<td>.4</td>
</tr>
<tr>
<td>0</td>
<td>24.98081</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>29.95207</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>35.39183</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>44.22326</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>56.83796</td>
<td></td>
</tr>
<tr>
<td>.1</td>
<td>65.25798</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>77.23849</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>85.62329</td>
<td></td>
</tr>
<tr>
<td>1.133</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>145.4806</td>
<td>22.48719</td>
<td>.6</td>
</tr>
<tr>
<td>0</td>
<td>22.48719</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>27.46931</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>32.42861</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>37.36192</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>50.79196</td>
<td></td>
</tr>
<tr>
<td>.1</td>
<td>59.24822</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>66.45775</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>73.6469</td>
<td></td>
</tr>
<tr>
<td>1.134</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>144.9825</td>
<td>21.2387</td>
<td>.8</td>
</tr>
<tr>
<td>0</td>
<td>21.2387</td>
<td></td>
</tr>
<tr>
<td>.01</td>
<td>24.98081</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>28.71144</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>36.13112</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>46.66129</td>
<td></td>
</tr>
<tr>
<td>.1</td>
<td>56.83796</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>62.85646</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>70.05393</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>1.135</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>20.36418</td>
</tr>
<tr>
<td>.01</td>
<td>24.48247</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>27.96634</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>34.89863</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>45.19923</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>54.42329</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>61.65458</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>66.45775</td>
<td></td>
</tr>
<tr>
<td>1.136</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>19.98925</td>
</tr>
<tr>
<td>.01</td>
<td>23.23577</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>25.72793</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>32.92315</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>44.46735</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>55.14818</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>59.24822</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>62.85646</td>
<td></td>
</tr>
<tr>
<td>1.137</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>19.98925</td>
</tr>
<tr>
<td>.01</td>
<td>22.98629</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>25.72793</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>32.92315</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>44.71137</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>54.9066</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>59.72979</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>63.33699</td>
<td></td>
</tr>
<tr>
<td>1.2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>95.23416</td>
</tr>
<tr>
<td>.01</td>
<td>109.7535</td>
<td></td>
</tr>
<tr>
<td>.02</td>
<td>114.6275</td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>125.1684</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>131.8287</td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>136.781</td>
<td></td>
</tr>
<tr>
<td>.2</td>
<td>138.766</td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>139.7593</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>1.201</td>
<td>1</td>
<td>145.9789</td>
</tr>
<tr>
<td>0</td>
<td>97.64432</td>
<td>.01</td>
</tr>
<tr>
<td>.05</td>
<td>121.3592</td>
<td>9.999999E-02</td>
</tr>
<tr>
<td>.2</td>
<td>138.766</td>
<td>.25</td>
</tr>
<tr>
<td>1</td>
<td>145.2316</td>
<td>80.83047</td>
</tr>
<tr>
<td>0</td>
<td>80.83047</td>
<td>.01</td>
</tr>
<tr>
<td>.05</td>
<td>129.8521</td>
<td>9.999999E-02</td>
</tr>
<tr>
<td>.2</td>
<td>140.5047</td>
<td>.25</td>
</tr>
<tr>
<td>1</td>
<td>146.7264</td>
<td>77.23849</td>
</tr>
<tr>
<td>0</td>
<td>77.23849</td>
<td>.01</td>
</tr>
<tr>
<td>.05</td>
<td>137.277</td>
<td>9.999999E-02</td>
</tr>
<tr>
<td>.2</td>
<td>144.2356</td>
<td>.25</td>
</tr>
<tr>
<td>1</td>
<td>145.4806</td>
<td>78.43568</td>
</tr>
<tr>
<td>0</td>
<td>78.43568</td>
<td>.01</td>
</tr>
<tr>
<td>.05</td>
<td>138.766</td>
<td>9.999999E-02</td>
</tr>
<tr>
<td>.2</td>
<td>144.2356</td>
<td>.25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>1.005</td>
<td>1.00</td>
<td>102.1882</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0001</td>
<td>28.71144</td>
<td></td>
</tr>
<tr>
<td>0.003</td>
<td>37.36192</td>
<td></td>
</tr>
<tr>
<td>0.005</td>
<td>41.04416</td>
<td></td>
</tr>
<tr>
<td>0.01</td>
<td>47.14814</td>
<td></td>
</tr>
<tr>
<td>0.015</td>
<td>54.42329</td>
<td></td>
</tr>
<tr>
<td>0.02</td>
<td>57.93054</td>
<td></td>
</tr>
<tr>
<td>0.03</td>
<td>64.05756</td>
<td></td>
</tr>
<tr>
<td>0.05</td>
<td>68.85565</td>
<td></td>
</tr>
<tr>
<td>9.999999E-02</td>
<td>76.04133</td>
<td></td>
</tr>
<tr>
<td>0.15</td>
<td>80.83047</td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>84.42454</td>
<td></td>
</tr>
<tr>
<td>0.25</td>
<td>95.23416</td>
<td></td>
</tr>
<tr>
<td>4.301</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20, 3,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>114.628</td>
<td>,31.1911</td>
<td>,.100000</td>
</tr>
<tr>
<td>0.000000E+00, 31.1911 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02, 33.6645 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02, 37.3619 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01, 42.2682 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01, 44.7114 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01, 47.1481 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01, 59.2482 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000 , 83.2262 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000 , 94.0303 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000 , 95.2342 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000 , 97.6443 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.302</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20, 3,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>115.849</td>
<td>,27.9663</td>
<td>,.200000</td>
</tr>
<tr>
<td>0.000000E+00, 27.9663 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02, 28.2148 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02, 28.7114 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01, 35.1453 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01, 36.8698 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01, 37.8538 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01, 44.7114 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000 , 59.2482 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000 , 76.0413 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000 , 85.6233 ,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000 , 90.4235 ,</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.303
20, 3,
115.360, 24.9808, 0.400000, 
11
0.000000E+00, 24.9808, 
0.300000E-02, 26.2257, 
0.500000E-02, 27.4693, 
0.100000E-01, 29.4560, 
0.150000E-01, 32.4286, 
0.200000E-01, 33.6645, 
0.500000E-01, 41.0442, 
10
0.100000, 52.0037, 
0.150000, 68.8557, 
0.200000, 73.6469, 
0.250000, 78.4357, 
4.304
20, 3,
115.849, 24.9808, 0.600000, 
10
0.000000E+00, 24.9808, 
0.500000E-02, 26.2257, 
0.100000E-01, 27.4693, 
0.150000E-01, 29.9521, 
0.200000E-01, 32.4286, 
0.500000E-01, 39.8184, 
0.100000, 47.1481, 
0.150000, 59.2482, 
0.200000, 64.0576, 
0.250000, 68.8557, 
4.305
20, 3,
114.628, 23.7346, 0.800000, 
11
0.000000E+00, 23.7346, 
0.300000E-02, 24.3579, 
0.500000E-02, 24.9808, 
0.100000E-01, 26.2257, 
0.150000E-01, 28.7114, 
0.200000E-01, 29.9521, 
0.500000E-01, 37.3619, 
0.100000, 44.1012, 
0.150000, 53.2141, 
0.200000, 58.0436, 
0.250000, 62.8565,
<table>
<thead>
<tr>
<th>Value</th>
<th>1st Column</th>
<th>2nd Column</th>
<th>3rd Column</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>3</td>
<td>115.849</td>
<td>23.7346</td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>23.7346</td>
<td>1.00000</td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>23.9839</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>24.9808</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>25.9769</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>28.0906</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>29.9521</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000E-01</td>
<td>36.1311</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.307</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>115.849</td>
<td>23.2358</td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>23.2358</td>
<td>1.20000</td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>23.7346</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>24.3579</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>24.9808</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>26.2257</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>28.7114</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>36.1311</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>43.4906</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>50.7920</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>55.6312</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>58.6460</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.308</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>115.849</td>
<td>23.9839</td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>23.9839</td>
<td>1.40000</td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>23.9839</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>24.3579</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>25.6034</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>26.2257</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>29.3319</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>37.3619</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>45.9305</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>53.2140</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>53.2141</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>56.8380</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>Column 1</td>
<td>Column 2</td>
<td>Column 3</td>
</tr>
<tr>
<td>-------</td>
<td>----------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td>4.201</td>
<td>20, 2,</td>
<td>117.071</td>
<td>39.8184</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>0.000000E+00, 39.8184</td>
<td>0.300000E-02, 44.7114</td>
</tr>
<tr>
<td></td>
<td>0.100000E-01, 49.5788</td>
<td>0.150000E-01, 56.8380</td>
<td>0.200000E-01, 61.6546</td>
</tr>
<tr>
<td></td>
<td>0.500000E-01, 76.0413</td>
<td>0.100000, 90.4235</td>
<td>0.150000, 97.6443</td>
</tr>
<tr>
<td></td>
<td>0.200000, 100.058</td>
<td>0.250000, 102.476</td>
<td>4.202</td>
</tr>
<tr>
<td></td>
<td>20, 2,</td>
<td>116.582</td>
<td>31.1911</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>0.000000E+00, 31.1911</td>
<td>0.300000E-02, 34.8986</td>
</tr>
<tr>
<td></td>
<td>0.100000E-01, 41.7788</td>
<td>0.150000E-01, 44.7114</td>
<td>0.200000E-01, 47.6348</td>
</tr>
<tr>
<td></td>
<td>0.500000E-01, 59.2482</td>
<td>0.100000, 77.9568</td>
<td>0.150000, 86.8225</td>
</tr>
<tr>
<td></td>
<td>0.200000, 91.1443</td>
<td>0.250000, 95.2342</td>
<td>4.203</td>
</tr>
<tr>
<td></td>
<td>20, 2,</td>
<td>117.071</td>
<td>27.4693</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>0.000000E+00, 27.4693</td>
<td>0.300000E-02, 28.7114</td>
</tr>
<tr>
<td></td>
<td>0.100000E-01, 33.6645</td>
<td>0.150000E-01, 37.3619</td>
<td>0.200000E-01, 39.8184</td>
</tr>
<tr>
<td></td>
<td>0.500000E-01, 48.3642</td>
<td>0.100000, 61.6546</td>
<td>0.150000, 76.0413</td>
</tr>
<tr>
<td></td>
<td>0.200000, 78.4357</td>
<td>0.250000, 80.8305</td>
<td></td>
</tr>
<tr>
<td>114.628</td>
<td>23.7346</td>
<td>0.800000</td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>23.7346</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>24.9808</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>26.9721</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>27.4693</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>29.4560</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>34.8986</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>37.3619</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>48.3642</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>56.8380</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>62.8565</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>67.6570</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.205</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| 115.849 | 22.9863 | 1.00000 |
| 0.000000E+00 | 22.9863 | |
| 0.300000E-02 | 24.4825 | |
| 0.500000E-02 | 24.9808 | |
| 0.100000E-01 | 26.2257 | |
| 0.150000E-01 | 27.4693 | |
| 0.200000E-01 | 32.4286 | |
| 0.500000E-01 | 43.4906 | |
| 0.100000 | 52.4880 | |
| 0.150000 | 59.2482 | |
| 0.200000 | 64.0576 | |
| 0.250000 | 66.4577 | |</p>
<table>
<thead>
<tr>
<th>Value1</th>
<th>Value2</th>
<th>Value3</th>
<th>Value4</th>
<th>Value5</th>
<th>Value6</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.207</td>
<td>20, 2,</td>
<td>115.849, 22.2376, 1.20000, 11</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00, 22.2376, 0.300000E-02, 23.7346, 0.500000E-02, 24.9808, 0.100000E-01, 26.2257, 0.150000E-01, 27.4693,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01, 32.4286, 0.500000E-01, 37.3619, 0.100000, 52.0037, 0.150000, 59.2482, 0.200000, 65.2580, 0.250000, 67.6570, 4.208, 20, 2, 115.849, 22.9863, 1.40000,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00, 22.9863, 0.300000E-02, 23.9839, 0.500000E-02, 26.9721, 0.100000E-01, 28.5873, 0.200000E-01, 29.9521, 0.500000E-01, 42.2682, 0.100000, 49.5788, 0.150000, 58.0436, 0.200000, 59.2482, 0.250000, 59.7298, 4.001, 1, 3, 114.628, 21.2387, 0.40000,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00, 21.2387, 0.300000E-02, 23.7346, 0.500000E-02, 26.9721, 0.100000E-01, 28.5873, 0.200000E-01, 36.1311, 0.500000E-01, 44.7114, 0.250000, 73.6469</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.002</td>
<td>1, 3,</td>
<td>115.849</td>
<td>20.2392</td>
<td>0.600000</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>--------</td>
<td>--------</td>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>20.2392</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>24.9808</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>26.2257</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>32.4286</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>33.6645</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>42.2682</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>55.6312</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>66.4577</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>68.8557</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>73.6469</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.003</td>
<td>1, 3,</td>
<td>117.071</td>
<td>19.9893</td>
<td>0.800000</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>--------</td>
<td>--------</td>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>19.9893</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>21.2387</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>24.4825</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>29.4560</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>30.4479</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>37.8538</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>49.5788</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>60.4519</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>62.8565</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>68.8557</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.004</td>
<td>1, 3,</td>
<td>116.338</td>
<td>19.9893</td>
<td>1.000000</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>--------</td>
<td>--------</td>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>19.9893</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>20.7390</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>22.3624</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>23.2358</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>27.4693</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>29.9521</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>39.8184</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>47.1481</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>56.8380</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>58.0436</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>61.6546</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1, 3,</td>
<td>117.071, 19.4892, 1.20000</td>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>------</td>
<td>--------------------------</td>
<td>----</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>19.4892</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>20.4891</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>21.2387</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>24.4825</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>29.7041</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>39.8184</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.10000</td>
<td>49.5788</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.15000</td>
<td>58.0436</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.20000</td>
<td>59.2482</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.25000</td>
<td>61.6546</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.006</td>
<td>1, 3,</td>
<td>117.071, 18.7390, 1.40000</td>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>18.7390</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>19.9893</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>20.9889</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>21.9879</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>27.9663</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>39.8184</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.10000</td>
<td>52.0037</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.15000</td>
<td>59.2482</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.20000</td>
<td>59.2482</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.25000</td>
<td>61.6546</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.101</td>
<td>20, 1,</td>
<td>117.071, 29.9521, 0.100000</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>29.9521</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>39.8184</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>42.7574</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>53.2141</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>59.2482</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>73.6469</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.10000</td>
<td>90.4235</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.15000</td>
<td>97.6443</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.20000</td>
<td>100.058</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.25000</td>
<td>102.476</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>4.102</td>
<td>20, 1,</td>
<td>114.628</td>
<td>24.9808</td>
<td>0.200000</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.000000E+00, 24.9808</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.300000E-02, 29.9521</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.500000E-02, 32.4286</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.100000E-01, 37.3619</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.150000E-01, 44.7114</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.200000E-01, 48.3642</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.500000E-01, 56.8380</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.10000, 77.2385</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.15000, 88.0222</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.20000, 90.4235</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.25000, 92.8273</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.103</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20, 1,</td>
<td>115.116</td>
<td>21.2387</td>
<td>0.400000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.000000E+00, 21.2387</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.300000E-02, 27.4693</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.500000E-02, 29.4560</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.100000E-01, 32.9231</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.150000E-01, 39.8184</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.200000E-01, 43.4906</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.500000E-01, 52.0037</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.10000, 64.0576</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.15000, 73.6469</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.20000, 76.0413</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.25000, 80.8305</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.104</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20, 1,</td>
<td>114.628</td>
<td>21.9879</td>
<td>0.600000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.000000E+00, 21.9879</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.300000E-02, 24.2332</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.500000E-02, 26.2257</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.100000E-01, 32.4286</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.150000E-01, 34.8986</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.200000E-01, 41.0442</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.500000E-01, 56.8380</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.10000, 65.2580</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.15000, 74.8442</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.20000, 76.0413</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.25000, 78.4357</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.105
20, 1,
119.030 , 21.9879 , 0.800000 ,
0.000000E+00, 21.9879 ,
0.300000E-02, 24.9808 ,
0.500000E-02, 26.2257 ,
0.100000E-01, 22.4872 ,
0.150000E-01, 37.3619 ,
0.200000E-01, 43.0019 ,
0.500000E-01, 61.6546 ,
0.10000, 77.2385 ,
0.15000, 83.2262 ,
0.25000, 85.6233 ,
5.001
20, 5,
115.238 , 24.9808 , 0.100000 ,
0.000000E+00, 24.9808 ,
0.300000E-02, 31.1911 ,
0.500000E-02, 37.3619 ,
0.100000E-01, 43.4906 ,
0.150000E-01, 50.7920 ,
0.200000E-01, 52.0037 ,
0.500000E-01, 62.8565 ,
0.10000 , 83.2262 ,
0.15000 , 94.0303 ,
0.20000 , 97.6443 ,
0.25000 , 101.266 ,
5.002
20, 5,
115.238 , 22.4872 , 0.200000 ,
0.000000E+00, 22.4872 ,
0.300000E-02, 22.6872 ,
0.500000E-02, 23.1110 ,
0.100000E-01, 23.9839 ,
0.150000E-01, 36.1311 ,
0.200000E-01, 38.5910 ,
0.500000E-01, 45.9305 ,
0.10000 , 58.6460 ,
0.15000 , 74.8442 ,
0.20000 , 84.4245 ,
0.25000 , 90.4235 ,
<p>| | | | | | | | | | | | | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>5.003</td>
<td>20</td>
<td>5</td>
<td>115.238</td>
<td>21.238</td>
<td>0.400000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0000000E+00</td>
<td>21.238</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>21.4387</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>21.6134</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>28.0906</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>29.9521</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>34.8986</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td></td>
<td>44.7114</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td></td>
<td>55.6312</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td></td>
<td>67.6570</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td></td>
<td>74.8442</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.004</td>
<td>20</td>
<td>5</td>
<td>115.238</td>
<td>20.2392</td>
<td>0.600000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0000000E+00</td>
<td>20.2392</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>20.6141</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>23.7346</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>24.6071</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>29.9521</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td></td>
<td>38.5910</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td></td>
<td>48.3642</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td></td>
<td>56.8380</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td></td>
<td>61.6546</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.005</td>
<td>20</td>
<td>5</td>
<td>114.628</td>
<td>19.9893</td>
<td>0.800000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0000000E+00</td>
<td>19.9893</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>19.9999</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>20.2392</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>21.2387</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>23.1110</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>23.7346</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>28.7114</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td></td>
<td>36.7467</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td></td>
<td>44.7114</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td></td>
<td>50.7920</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td></td>
<td>55.6312</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x</td>
<td>y1</td>
<td>y2</td>
<td>y3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>----------</td>
<td>----------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>19.3642</td>
<td>19.3642</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>19.9893</td>
<td>19.9893</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>20.9437</td>
<td>20.9437</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>21.2387</td>
<td>21.2387</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>22.4872</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>22.7368</td>
<td>22.7368</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>28.7114</td>
<td>28.7114</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>36.1311</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>39.8184</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>50.1856</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>53.8189</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.007</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>19.7393</td>
<td>19.7393</td>
<td>1.20000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>19.9893</td>
<td>19.9893</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>20.2392</td>
<td>20.2392</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>21.2387</td>
<td>21.2387</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>21.8631</td>
<td>21.8631</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>22.4872</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>28.7114</td>
<td>28.7114</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>34.8986</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>44.7114</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>48.3642</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>50.7920</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.008</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000000E+00</td>
<td>19.7393</td>
<td>19.7393</td>
<td>1.40000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.300000E-02</td>
<td>19.9893</td>
<td>19.9893</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-02</td>
<td>20.6141</td>
<td>20.6141</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000E-01</td>
<td>21.2387</td>
<td>21.2387</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000E-01</td>
<td>21.6134</td>
<td>21.6134</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000E-01</td>
<td>22.4872</td>
<td>22.4872</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.500000E-01</td>
<td>28.0906</td>
<td>28.0906</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.100000</td>
<td>34.8986</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.150000</td>
<td>43.4906</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.200000</td>
<td>46.1481</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.250000</td>
<td>52.0037</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix B
Listing of Liquid Jet Model

B.1 Liquid_jet.f90

MODULE JetIntrType
  IMPLICIT NONE

  INTEGER, PARAMETER :: sdk = selected_real_kind(13,307)
  INTEGER, PARAMETER :: sik = kind(1000000)
END MODULE JetIntrType

MODULE JetDataIn

  USE JetIntrType
  ! USE cobra_prop

  REAL(sdk), ALLOCATABLE :: lj_pres(:), lj_rlj(:), lj_tlj_av (:), lj_dy (:), lj_dyy (:)
  REAL(sdk), ALLOCATABLE :: lj_visc (:), lj_st (:), lj_kf (:), lj_pr (:), lj_edmy (:)
  REAL(sdk), ALLOCATABLE :: lj_vslj (:), lj_vdt (:), lj_pres_old (:), lj_x_old (:)
  REAL(sdk), ALLOCATABLE :: lj_aclj (:), lj_axlj (:), lj_id (:), lj_cp (:), lj_kidrag (:)
  REAL(sdk), ALLOCATABLE :: lj_ev (:), lj_uvd (:), lj_vi (:), lj_hhil (:), lj_b_old (:)
  REAL(sdk), ALLOCATABLE :: lj_ulj (:), lj_vlj (:), lj_glj (:), lj_aylj (:), lj_slj (:)
  REAL(sdk), ALLOCATABLE :: lj_uplj (:), lj_vlj_old (:), lj_tlj (:), lj_llj (:)
  REAL(sdk), ALLOCATABLE :: lj_t1lj_old (:), lj_is (:), lj_area_av (:), lj_vglj (:)
  REAL(sdk), ALLOCATABLE :: lj_i1lj_old (:), lj_z (:), lj_b (:), lj_x (:), lj_edh (:)
  REAL(sdk), ALLOCATABLE :: lj_edm (:), lj_rholj (:), lj_hil (:), lj_vol (:), lj_tsat (:)
  REAL(sdk), ALLOCATABLE :: lj_int (:), lj_ivint (:), lj_vif (:), lj_dhodi (:)
  REAL(sdk), ALLOCATABLE :: lj_volx (:), lj_voly (:), lj_drhodp (:), lj_uglj (:)
  & lj_u0 (:), lj_v0 (:), lj_c0 (:), lj_d0 (:)
  & lj_r (:), lj_rx (:), lj_ry (:), lj_dyx (:), lj_edmx (:)
  & lj_uslj (:), lj_ii (:), lj_g (:), lj_mass_in (:), lj_mass_out (:)
  & lj_en_in (:), lj_en_out (:), lj_qil (:), lj_giv (:), lj_difdp (:)
  & lj_rv_old (:), lj_rv (:), lj_cl (:), lj_c2 (:), lj_d1 (:), lj_d2 (:)
  & lj_gas (:), lj_gas_old (:), lj_edg (:), lj_if (:), lj_ig (:)
  & lj_dtmt (:), lj_tvb (:), lj_area (:), lj_eerr (:)
  & lj_perr (:), lj_uerr (:), lj_verr (:), lj_dqildp (:), lj_dhildp (:)
  & lj_difhdx (:), lj_difhrdt (:), lj_difmx (:), lj_dirmrst (:)
  & lj_pdif (:), lj_edif (:), lj_pgas (:), lj_pgas_old (:), lj_pstm (:)
  & lj_pstm_old (:), lj_tdp (:), lj_tdp_old (:), lj_hfdp (:), lj_hgd (:)
  & lj_xcond (:), lj_r (:), lj_i_rst (:), lj_ppa_rst (:)
  & lj_pps_rst (:), lj_tdp_rst (:), lj_u_rst (:), lj_v_rst (:), lj_gas_rst (:)


INTEGER(sik), ALLOCATABLE :: lj_ipvt(:)

! Arrays for 1-D Jet model!

REAL(sdk), ALLOCATABLE :: lj_u(:), lj_a(:), lj_a_old(:), lj_t(:), lj_i(:) &
lj_s(:), lj_g(:), lj_u_old(:), lj_t_old(:) &
lj_i_old(:), lj_tv(:), lj_volv(:)
REAL(sdk) :: lj_ifc, lj_igc, ttime
REAL(sdk) :: lj_pi=3.14159265358979_sdk
REAL(sdk) :: lj_gc=32.174_sdk ! Ideal gas constant
REAL(sdk) :: lj_rgas = 10.73_sdk ! psi-ft^3/(lbmol-R)
REAL(sdk) :: lj_mgas = 28.97_sdk ! Molecular weight of gas lbm/lbm
REAL(sdk) :: lj_sin2sft=144.0_sdk ! converts ft^2 -> in^2
REAL(sdk) :: lj_psft2btu=0.18273_sdk ! converts psia-ft^3/lbm -> Btu/lbm
REAL(sdk) :: lj_re_crit=2500.0 ! Critical Reynolds number
REAL(sdk) :: lj_re_tran=4000.0 ! Turbulent transition Re
REAL(sdk) :: lj_cnvlg=0.001

REAL(sdk) :: lj_dx, lj_dylj, lj_aclj, lj_axlj, lj_aylj, lj_slj, lj_pres,
            lj_rlj, lj_rxb, lj_ryi, lj_ryo, lj_iyi, lj_ivin,
            lj_rhovin, lj_dlj, lj_rvb, lj_lj, eerr, perr, berr
REAL(sdk) :: lj_difhxdt_max, lj_difhrdt_max, lj_difmxdt_max, lj_difmrdt_max,
            lj_difmin, lj_pdifmin
INTEGER(sik) :: lj_nx, lj_nylj, lj_nlj, lj_nvb, lj_n, lj_ny, lj_k,
                lj_l, lj_m, lj_nd, lj_lld, lj_lw, lj_lwu, lj_lldi,
                band, lj_count, lj_countmax

! lj_dx = Axial cell length
! lj_dylj = Radial cell length for liquid jet
! lj_aclj = Continuity cell area for liquid jet
! lj_axlj = Axial mom cell cross-sectional area for liquid jet
! lj_aylj = Transverse mom cell X-sectional area for liquid jet
! lj_slj = Liquid jet source term
! lj_pres = Pressure
! lj_rlj = Jet radius
! lj_dt = Time step size
! lj_viscv = Vapor viscosity
! lj_kviscv = Vapor kinematic viscosity
! lj_gc = Gravitational constant
! lj_nylj = Number of radial computational cells for liquid jet
! lj_k = Number of axial computational cells
! lj_nyb = Total number of radial computational cells
! lj_klj = Liquid jet axial velocity
! lj_vlj = Liquid jet transverse velocity
! lj_glj = Condensation mass transfer coefficient
! lj_cplj = Liquid jet specific heat
! lj_klj = Liquid jet thermal conductivity
**CONTAINS**

**SUBROUTINE** SetData

|     lj_props(10) | drdp, drdi, dtdp, ltdi, lj_dpin, dvdp, dvdi, lj_stlj, &     |
| lj_stv, lj_rholjin |

**IMPLICIT** NONE

**REAL(sdk)** :: lj_props(10), drdp, drdi, dtdp, ltdi, lj_dpin, dvdp, dvdi, lj_stlj, &     |
| lj_stv, lj_rholjin |

**INTEGER(sik)** :: lj_crit, lj_flag, iregn, lj_iphase, i, j, k, instat, q

**Set the dimensions for arrays in the vapor boundary layer and liquid jet**

| lj_nvb=lj_nx*lj_nyvb |
| lj_n1=lj_nx*lj_nylj |
| lj_ny=lj_nyvb+lj_nylj |
| lj_n=lj_nx*lj_ny |
| lj_nd=(lj_nx+2)*(lj_ny+2) |
| lj_l=lj_n+lj_ny |
| lj_lid=(lj_nx+3)*(lj_ny+2) |
| lj_m=lj_n+lj_nx |
| lj_md=(lj_nx+2)*(lj_ny+3) |
| lj_wu=2_sik*lj_ny+1 |
| lj_wl=4_sik*lj_ny |
| lj_w=lj_wu+lj_wl+1 |
| lj_lid=2_sik*lj_wl+lj_wu+1 |
| lj_k=lj_n+lj_n+lj_n |

**Allocate the arrays in the vapor boundary layer and liquid jet**

**ALLOCATE** (lj_rholj(lj_nx), lj_pres(lj_n), lj_vlj_old(lj_md), lj_rholj(lj_md))

**ALLOCATE** (lj_tlj(lj_n), lj_tlj_av(lj_nx), lj_axlj(lj_n), lj_ulj(lj_l))
ALLOCATE (lj_ac1lj(lj_n), lj_vl1j(lj_m), lj_ul1j_old(lj_ld), lj_pres_old(lj_nd))
ALLOCATE (lj_g1j(lj_n), lj_aylj(lj_m), lj_sl1j(lj_nd), lj_area_av(lj(nx))
ALLOCATE (lj_id(lj_nx), lj_u1vrd(lj_nx), lj_11lj_old(ljNd), lj_v(lj_nx))
ALLOCATE (lj_1lj(lj_n), lj_edh(lj_n), lj_edm(lj_n), lj_h1il(lj_nx), lj_x(lj_k))
ALLOCATE (lj_b(lj_k), lj_is(lj_nylj), lj_ipvt(lj_k), lj_x_old(lj_k))
ALLOCATE (lj_st(lj_n), lj_visc(lj_n), lj_kf(lj_n), lj_dy(lj_n), lj_cp(lj_n))
ALLOCATE (lj_drhodi(lj_n), lj_kidrag(lj_l), lj_dyy(lj_m), lj_vif(lj_nx))
ALLOCATE (lj_iint(lj_nx), lj_ivint(lj_nx), lj_vol(lj_n), lj_hhil(lj_nx))
ALLOCATE (lj_volx(lj_l), lj_vol(y(lj_m), lj_u0(lj_l), lj_c0(lj_l), lj_d0(lj_m))
ALLOCATE (lj_vslj(lj_m), lj_edmy(lj_m), lj_pr(lj_n), lj_vglj(lj_m), lj_v0(lj_m))
ALLOCATE (lj_r(lj_n), lj_rx(lj_l), lj_ry(lj_m), lj_drhodp(lj_n), lj_dyx(lj_l))
ALLOCATE (lj_mass_in(lj_n), lj_mass_out(lj_n), lj_rv_old(lj_n), lj_rv(lj_n))
ALLOCATE (lj_diffdp(lj_n), lj_edif(lj_n), lj_1lj_g(lj_n), lj_edq(lj_n))
ALLOCATE (lj_en_out(lj_n), lj_qil(lj_n), lj_qiv(lj_n), lj_pdif(lj_n))
ALLOCATE (lj_en_in(lj_n), lj_qil(lj_n), lj_qiv(lj_n), lj_pdif(lj_n))
ALLOCATE (lj_if(lj_n), lj_ig(lj_n), lj_dtmt(lj_nx), lj_tvav(lj_nx))
ALLOCATE (lj_vb_area(lj_nx), lj_tsat(lj_n), lj_gas_old(lj_n), lj_vdt(lj_m))
ALLOCATE (lj_uerr(lj_l), lj_verr(lj_m), lj_verr(lj_n), lj_eerr(lj_n))
ALLOCATE (lj_dq1dp(lj_n), lj_dhildp(lj_n), lj_difhdrt(lj_n), lj_difhrdt(lj_n))
ALLOCATE (lj_b_old(lj_k), lj_berr(lj_k), lj_p_rst(lj_n), lj_i_rst(lj_n))
ALLOCATE (lj_difmxdt(lj_n), lj_difmrdt(lj_n), lj_ppa_rst(lj_n))
ALLOCATE (lj_pps_rst(lj_n), lj_gas_rst(lj_n), lj_pstm(lj_n), lj_tdp(lj_n))
ALLOCATE (lj_tdp_rst(lj_n), lj_u_rst(lj_l), lj_v_rst(lj_m), lj_xcond(lj_nx))
ALLOCATE (lj_pvgs_old(lj_n), lj_pstm_old(lj_n), lj_tdp_old(lj_n))
ALLOCATE (lj_hfdp(lj_n), lj_hgdp(lj_n), lj_pgas(lj_n))

!**************************************************************************
!   Allocate the arrays for the one-dimensional liquid jet continuity equation
!**************************************************************************

IF (band) THEN
   ALLOCATE (lj_z(lj_ldi, lj_k))
ELSE
   ALLOCATE (lj_z(lj_k, lj_k))
ENDIF

! Clear all of the arrays

lj_iljin   = 0.0
lj_props   = 0.0
lj_ac1lj   = 0.0
lj_aylj    = 0.0
lj_axlj    = 0.0
lj_ul1j    = 0.0
lj_vl1j    = 0.0
lj_sl1j    = 0.0
lj_dylj    = 0.0
lj_dyvb    = 0.0
lj_dx      = 0.0
lj_pres    = 0.0
lj_gas     = 0.0
lj_gas_old = 0.0
lj_g1j     = 0.0
lj_u       = 0.0
lj_t       = 0.0
lj_i       = 0.0
lj_s       = 0.0
lj_g = 0.0

! Set all of the old time arrays to zero as well
!
lj_ulsj_old = 0.0
lj_vlsj_old = 0.0
lj_u_old = 0.0
lj_v_old = 0.0
lj_i_old = 0.0
lj_x_old = 0.0
lj_z = 0.0
lj_x = 0.0
lj_b = 0.0
lj_vslj = 0.0
lj_vglj = 0.0
lj_uslj = 0.0
lj_uglj = 0.0

!#############################################################################
! Calculate Initial properties
CALL WPINIT(0,0,0,12,INSTAT)

! Calculate saturation pressure if given a saturation temperature
!*****************************************************************************
IF(lj_pin.eq.0.0) THEN
  CALL psat_of_t_vapor(lj_tstm,lj_pin,lj_dpin,lj_crit,lj_flag)
ENDIF

! Find the liquid properties
!]*****************************************************************************
! Find the input enthalpy of the liquid jet
! lj_iphase=1
CALL fl9_hpt1(lj_pin,lj_tin,lj_iljin,lj_iphase)

! Find the initial jet density, surface tension, thermal cond., and viscosity
CALL RHOTLQ(lj_pin,lj_iljin,lj_props,lj_rholjin,lj_tin, &
  drdp,drdi,dtdp,dtdi,dvdp,dvdi,lj_stlj,lj_klj, &
  lj_visclj,iregn,lj_crit,lj_flag)

lj_drhodp=drdp
lj_drhodi=drdi
lj_cp=1.0_sdk/dtdi
lj_ifc=lj_props(3)
lj_igc=lj_props(4)
lj_visc=lj_visclj
lj_st=lj_stlj
lj_kf=lj_klj

!*****************************************************************************
! Find the vapor properties
!*****************************************************************************
! Find the enthalpy of the vapor boundary
! lj_iphase=2
CALL fl9_hpt1(lj_pin, lj_tstm, lj_invin, lj_iphase)

! Find the vapor density, surface tension, thermal cond. and viscosity
!
CALL RHOTVP(lj_pin, lj_invin, lj_props, lj_rhovin, lj_tstm, &
  drdp, drdi, dtdp, dtdi, dvdp, dvdi, lj_stv, lj_kv, &
  lj_viscv, iregn, lj_crit, lj_flag)
!
!******************************************************************************
!
!     Now that we know the initial properties, initiate the fluid arrays
!
!******************************************************************************
!
!     Initialize arrays
!
1j_dx=lj_length/lj_nx
!
1j_rlj=0.5*1j_din
1j_rvb=0.5*1j_dvb
1j_dyvb=0.5*(1j_dvb-1j_din)/lj_nyvb
1j_dylj=0.5*1j_din/lj_nylj
!
!
!     Set Density array
1j_rholj=1j_rholjin
DO i=0,lj_nx+1
  DO j=lj_nylj+1,lj_ny+1
    q=(j+1)+i*(lj_ny+2)
    1j_rholj(q)=lj_rhovin
  ENDDO
ENDDO
!
!     Set enthalpy array
1j_ilj_old=1j_iljin
DO i=0,lj_nx+1
  DO j=lj_nylj+1,lj_ny+1
    q=(j+1)+i*(lj_ny+2)
    1j_ilj_old(q)=lj_ivin
    !
    1j_gas_old(q)=lj_gasin
  ENDDO
ENDDO
!
!     Set property arrays

DO i=1,lj_nx
  DO j=1,lj_ny
    k=j+(i-1)*lj_ny
    q=(j+1)+i*(lj_ny+2)
    1j_ilj(k)=1j_ilj_old(q)
    !
    1j_gas(k)=1j_gas_old(q)
  ENDDO
ENDDO
lj_cp(k) = 1.0_sdk/dtdi
lj_visc(k) = lj_viscv
lj_kf(k) = lj_kv
lj_st(k) = lj_stv
ENDDO
ENDDO
lj_pr = lj_visc * lj_cp / lj_kf

DO i = 1, lj_nx
  DO j = 1, lj_ny
    k = j + (i - 1) * lj_ny
    q = (j + 1) + i * (lj_ny + 2)
    lj_edm(k) = lj_visc(k) / lj_rholj(q)
  ENDDO
ENDDO

lj_edm = 0.0_sdk
lj_edh = lj_edm / lj_pr   != lj_edm / lj_pr != lj_kf(k) / (lj_rholj(q) * lj_cp(k))
lj_id = lj_ig
lj_is = lj_if

!     Set initial 1-D calculation arrays
!
jl_1ljin = 18.1086928779532_sdk
lj_a = 0.25 * lj_pi * lj_din * lj_din
lj_a_old = lj_a
lj_u = lj_uin
lj_u_old = lj_u
lj_t = lj_tin
lj_t_old = lj_t
lj_i = lj_iljin
lj_i_old = lj_i
lj_int = lj_1ljin
lj_ivint = lj_ig
lj_vin = 0.0_sdk
!
!     Set pressure array
lj_pres = lj_pin
lj_pres_old = lj_pin

!
!     Set liquid jet arrays
!
DO i = 0, lj_nx + 2
  DO j = 0, lj_ny + lj
    q = (j + 1) + i * (lj_ny + 2)
    lj_ulj_old(q) = lj_uin
  ENDDO
ENDDO
DO i = 1, lj_nx + 1
  DO j = 1, lj_ny
    k = j + (i - 1) * lj_ny
    q = (j + 1) + i * (lj_ny + 2)
    lj_ulj(k) = lj_ulj_old(q)
  ENDDO
ENDDO
ENDDO

lj_vi = 0.0_sdk
lj_dt=lj_dx/4800.0_sdk
lj_dt=lj_dx/lj_uin
IF (lj_dt.gt.lj_tmax) THEN
   lj_dt=lj_tmax
ENDIF

DO j=1,lj_nylj
   q=(j+1)+i*(lj_ny+2)
   lj_slj(q)=lj_uin*lj_rholj(q)/lj_dx
ENDDO

lj_tlj=lj_tin
lj_tlj_old=lj_tlj
lj_ilj=lj_iljin
lj_ilj_old=lj_ilj

END SUBROUTINE SetData

END MODULE JetDataIn

MODULE JetSetEqn

!*****************************************************************************!
!  This module contains the models for a 1-D liquid jet to find the radius    !
!  of the jet for use in the 2-D problem. It then sets the coefficient       !
!  matrix Z and the RHS Matrix x for the 2-D condensing liquid jet           !
!                                                                            !
!  Programmed by Frank Buschman                                              !
!     rev 1. 4/2007                                                          !
!     rev 2. 5/2007                                                          !
!*****************************************************************************!

USE JetIntrType
USE JetDataIn

CONTAINS

SUBROUTINE OneDJet

!*****************************************************************************!
!  This subroutine will solve the 1-D jet continuity equation to find the     !
!  radius of the liquid jet for use in the 2-D calculations                   !
!                                                                            !
!  Programmed by Frank Buschman                                              !
!     rev. 2 04/07/2007                                                        !
!*****************************************************************************!
! IMPLICIT NONE
!
INTEGER(sik) :: i,j,k
REAL(sdk) :: ui,vid,div,ain,ui_1

ain=0.25*lj_pi*lj_din*lj_din
!
j=lj_nylj
DO i=1,lj_nx
  k=j+(i-1)*lj_ny
  IF (i.eq.1) THEN
    ui=0.5*(lj_uin+lj_u(i))
    vid=lj_dx+ui*lj_dt
    div=1.0_sdk/vid
    lj_a(i)=(lj_a_old(i)*lj_dx+ain*lj_uin*lj_dt+(lj_a_old(i)*ui* &
               (lj_if(k)-lj_i(i)))/(lj_ig(k)-lj_i(i))*lj_dt)*div
  ENDIF
  IF (i.eq.2) THEN
    ui=0.5*(lj_u(i-1)+lj_u(i))
    ui_1=0.5*(lj_uin+lj_u(i-1))
    vid=lj_dx+ui*lj_dt
    div=1.0_sdk/vid
    lj_a(i)=(lj_a_old(i)*lj_dx+lj_a_old(i-1)*ui_1*lj_dt+(lj_a_old(i)* &
               ui*(lj_if(k)-lj_i(i)))/(lj_ig(k)-lj_i(i))*lj_dt)*div
  ENDIF
  IF (i.ne.1 .and. i.ne.2) THEN
    ui=0.5*(lj_u(i-1)+lj_u(i))
    ui_1=0.5*(lj_u(i-2)+lj_u(i-1))
    vid=lj_dx+ui*lj_dt
    div=1.0_sdk/vid
    lj_a(i)=(lj_a_old(i)*lj_dx+lj_a_old(i-1)*ui_1*lj_dt+(lj_a_old(i)* &
               ui*(lj_if(k)-lj_i(i)))/(lj_ig(k)-lj_i(i))*lj_dt)*div
  ENDIF
ENDDO
!
!     Find the jet radius from the area
!     DO i=1,lj_nx
!        WRITE(15,*) lj_a(i), lj_a_old(i), lj_u(i)
!     ENDDO
lj_rlj=(lj_a/lj_pi)**0.5
lj_a_old=lj_a
ENDSUBROUTINE OneDJet

SUBROUTINE TwoJetLHS

*****************************************************************************!
!   This subroutine Develops the left hand side of the equations for the     !
!        2-D jet                                                              !
!                                                                             !
!     Programmed by Frank Buschman                                            !
!       rev. 5  7/24/2007                                                     !
*****************************************************************************!
!
IMPLICIT NONE
!
INTEGER(sik) :: i,j,k,k1,k2,k3,k4,k5,k6,k7,k8,k9,k10,q
!
REAL(sdk) ::
! Set the matrix for the energy equation
DO i=1,lj_nx
  DO j=1,lj_ny
    k=j+(i-1)*lj_ny
    q=(j+1)+i*(lj_ny+2)
    IF (band) THEN
      k1=lj_w
      k2=lj_w-1
      k3=lj_w-1+2
      k4=lj_w-1-2
      k5=lj_w-1+2*lj_ny
      k6=lj_w-1-2*lj_ny
    ELSE
      k1=k+k+k
      k2=k+k+k
      k3=k+k+k
      k4=k+k+k
      k5=k+k+k
      k6=k+k+k
      k7=k+k+k
      k8=k+k+k
      k9=k+k+k
      k10=k+k+k
    ENDIF
    IF (lj_vlj_old(q+1+(i-1)).gt.0.0_sdk) THEN
      lj_ryo=lj_rholj(q)
      lj_iyo=lj_ilj_old(q)
    ELSE
      lj_ryo=lj_rholj(q+1)
      lj_iyo=lj_ilj_old(q+1)
    ENDIF
    IF(lj_ulj_old(q).ge.0.0_sdk) THEN
      lj_rxt=lj_rholj(q-(lj_ny+2))
      lj_ixt=lj_ilj_old(q-(lj_ny+2))
    ELSE
      lj_rxt=lj_rholj(q)
      lj_ixt=lj_ilj_old(q)
    ENDIF
    IF(lj_ulj_old(q).ge.0.0_sdk) THEN
      lj_rxb=lj_rholj(q)
      lj_ixb=lj_ilj_old(q+(lj_ny+2))
    ELSE
      lj_rxb=lj_rholj(q+(lj_ny+2))
      lj_ixb=lj_ilj_old(q+(lj_ny+2))
    ENDIF
  ENDDO
ENDO
lj_ryi = lj_rholj(q-1)
lj_iyo = lj_ilj_old(q)
lj_iyi = lj_ilj_old(q-1)

IF (j.eq.lj_nylj) THEN
lj_ryo = lj_rholj(q)
lj_iyo = lj_ilj_old(q)
lj_iyi = lj_ilj_old(q-1)
ENDIF

IF (j.eq.lj_nylj+1.or.j.eq.1) THEN
lj_ryi = lj_rholj(q)
lj_iyi = lj_ilj_old(q)
ENDIF

General form of the energy equation

lj_z(k1,k+k+k)=(lj_rholj(q)+lj_ilj_old(q)*lj_drhodi(k))*lj_r(k)*lj_dx*lj_dy(k)*lj_dy(k)+lj_r(k)*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt*(lj_rxb*lj_ixb*lj_c2(k+lj_ny)*(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)-lj_rxt*lj_ixt*lj_c2(k)*(1.0_sdk+lj_kidrag(k)*lj_dt))
lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dt*lj_rholj(q)*lj_ilj_old(q)*(lj_d2(k+1+(i-1))-lj_d2(k+(i-1)))+0.5_sdk*(lj_d2(k+(i-1))+lj_d1(k+(i-1))+lj_d1(k+1+(i-1))-lj_d0(k+(i-1)))*((lj_rholj(q)*(lj_iyo-lj_iyi)+lj_ilj_old(q)*(lj_ryo-lj_ryi))*lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dt+lj_rholj(q)*lj_ilj_old(q)*lj_dx*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt)

lj_z(k2,k+k+k-1)=(lj_ilj_old(q)*lj_drhodp(k)-lj_psft2btu)*lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dy(k)+lj_r(k)*(lj_rxb*lj_ixb*(lj_c1(k+lj_ny)-lj_c0(k+lj_ny))*(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)-lj_rxt*lj_ixt*(lj_c1(k)+lj_c0(k))*(1.0_sdk+lj_kidrag(k)*lj_dt))*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt+lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dt*lj_rholj(q)*lj_ilj_old(q)*lj_dx*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt)

IF(j.eq.lj_nylj)THEN
lj_z(k1,k+k+k)=lj_z(k1,k+k+k)*lj_qil(k)/lj_vol(k)*lj_r(k)*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt*(1.0+lj_if(k+1)/(lj_ilj_old(q+1)-lj_if(k+1)))*lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dt

ENDIF

IF(j.eq.lj_nylj+1)THEN
lj_z(k2,k+k+k-1)=lj_z(k2,k+k+k-1)+lj_qil(k-1)/lj_vol(k)*lj_difdp(k)*lj_r(k)*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt

ENDIF
IF (j.ne.lj_ny) THEN

lj_z(k3,3_sik*(k+1)-1)=(lj_d0(k+1+(i-1))+lj_d1(k+1+(i-1)))*
((lj_rholj(q)*lj_iij_old(q)+0.5_sdk*lj_rholj(q)*lj_iyo-lj_iyi)+
0.5_sdk*lj_ilj_old(q)*(lj_ryo-lj_ryi))*
lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dt+
0.5_sdk*lj_rholj(q)*lj_ilj_old(q)*lj_dx*lj_dy(k)*lj_dt+
lj_dy(k)*lj_dy(k)*lj_dt)

lj_z(k7,3_sik*(k+1))=lj_d2(k+1+(i-1))*(lj_iij_old(q)+
0.5_sdk*lj_iij_old(q)+0.5_sdk*lj_ilj_old(q)*
(lj_ryo-lj_ryi))*lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dt+
0.5_sdk*lj_rholj(q)*lj_ilj_old(q)*lj_dx*lj_dy(k)*lj_dt)

IF (j.eq.lj_nylj) THEN

lj_z(k3,3_sik*(k+1)-1)=lj_z(k3,3_sik*(k+1)-1)-lj_qil(k)/lj_vol(k)*
(lj_iij_old(q)-lj_if(k+1))/(lj_iij_old(q)+lj_iij_old(q)+lj_if(k)+lj_if(k+1))*lj_difdp(k+1)*
lj_r(k)*lj_dy(k)*lj_dy(k)*lj_dx*lj_dx*lj_dt

ENDIF

IF (j.ne.1) THEN

lj_z(k4,3_sik*(k-1)-1)=(lj_d0(k+(i-1))-lj_d1(k+(i-1)))*
((lj_rholj(q)-lj_iij_old(q)-0.5_sdk*lj_rholj(q)*
lj_iyo-lj_iyi)+0.5_sdk*lj_ilj_old(q)*(lj_ryo-lj_ryi))*
lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dt-0.5_sdk*lj_rholj(q)*
lj_ilj_old(q)*lj_dx*lj_dy(k)*lj_dt)

lj_z(k8,3_sik*(k-1))=lj_d2(k+(i-1))*(0.5_sdk*lj_rholj(q)*
(lj_iyo-lj_ryi)+0.5_sdk*lj_iij_old(q)+lj_if(k+1))*lj_difdp(k+1)*
lj_r(k)*lj_dy(k)*lj_dy(k)*lj_dx*lj_dx*lj_dt

IF (j.eq.lj_nylj+1) THEN

lj_z(k8,3_sik*(k-1))=lj_z(k8,3_sik*(k-1))-lj_qil(k-1)/lj_vol(k)*
(lj_iij_old(q)/(lj_iij_old(q)-lj_if(k)))*
lj_r(k)*lj_dy(k)*lj_dy(k)*lj_dx*lj_dx*lj_dt

ENDIF

ENDIF

IF (i.ne.lj_nx) THEN

lj_z(k5,3_sik*(k+lj_ny)-1)=lj_r(k)*lj_rxb*
(lj_c0(k+lj_ny)+lj_c1(k+lj_ny))*
(lj_ixo*k+lj_joy)*
lj_ixb*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt

lj_z(k9,3_sik*(k+lj_ny))=lj_r(k)*lj_rxb*lj_ixb*
lj_c2(k+lj_ny)*(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)*
lj_dx*lj_dy(k)*lj_dy(k)*lj_dt

ENDIF

IF (i.ne.1) THEN

lj_z(k6,3_sik*(k-lj_ny)-1)=lj_r(k)*lj_rxt*(lj_c0(k)-lj_c1(k))*
(1.0_sdk+lj_kidrag(k)*lj_dt)*lj_ixt*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt

lj_z(k10,3_sik*(k-lj_ny))=-1.0_sdk*lj_r(k)*lj_rxt*lj_ixt*lj_c2(k)*
(1.0_sdk+lj_kidrag(k)*lj_dt)*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt

ENDIF
! At j=1, move the j-1 term to the j term
IF (j.eq.1) THEN
   lj_z(k2,k+k-1)=lj_z(k2,k+k+1)+(lj_d0(k+(i-1))-lj_d1(k+(i-1)))*
   (lj_rholj(q)*lj_i1lj_old(q)-0.5_sdk*lj_rholj(q)*
   (lj_iyo-lj_iyi)0.5_sdk*lj_i1lj_old(q)*(lj_ryo-lj_ryi))*
   lj_z(k1,k+k+1)=lj_z(k1,k+k+1)+lj_d2(k+(i-1))*((0.5_sdk*lj_rholj(q)*
   (lj_iyo-lj_iyi)+0.5_sdk*lj_i1lj_old(q)-
   lj_rholj(q)*lj_i1lj_old(q))*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt+0.5_sdk*lj_rholj(q)*
   lj_i1lj_old(q)*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt)
ENDIF

lj_z(k1,k+k)=lj_z(k1,k+k)+lj_d2(k+(i-1))*((0.5_sdk*lj_rholj(q)*
   (lj_iyo-lj_iyi)+0.5_sdk*lj_i1lj_old(q)-
   lj_rholj(q)*lj_i1lj_old(q))*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt)

ENDIF

! lj_z(k1,k+k)=1.0_sdk
ENDDO
ENDDO

! Set the matrix of the continuity equation
DO i=1,lj_nx
   DO j=1,lj_ny
      k=j+(i-1)*lj_ny
      q=(j+1)+i*(lj_ny+2)

      IF (band) THEN
         k1=k+k+k-1
         k2=k+k+k-1
         k3=k+k+k-1
         k4=k+k+k-1
         k5=k+k+k-1
         k6=k+k+k-1
         k7=k+k+k-1
         k8=k+k+k-1
         k9=k+k+k-1
         k10=k+k+k-1
      ELSE
         k1=k+k+k-1
         k2=k+k+k-1
         k3=k+k+k-1
         k4=k+k+k-1
         k5=k+k+k-1
         k6=k+k+k-1
         k7=k+k+k-1
         k8=k+k+k-1
         k9=k+k+k-1
         k10=k+k+k-1
      ENDIF

      IF (lj_vlj_old(q+1+(i-1)).gt.0.0_sdk) THEN
         lj_ryo=lj_rholj(q)
         lj_iyo=lj_i1lj_old(q)
      ELSE
         lj_ryo=lj_rholj(q+1)
         lj_iyo=lj_i1lj_old(q+1)
      ENDIF

      IF(lj_vlj_old(q+(i-1)).gt.0.0_sdk) THEN
! lj_ryi=lj_rholj(q-1)!
! lj_iyi=lj_ilj_old(q-1)!
ELSE
! lj_ryi=lj_rholj(q)
! lj_iyi=lj_ilj_old(q)
ENDIF

IF(lj_ulj_old(q).ge.0.0_sdk)THEN
   lj_rxt=lj_rholj(q-(lj_ny+2))
   lj_ixt=lj_ilj_old(q-(lj_ny+2))
ELSE
   lj_r!t=lj_rholj(q)
   lj_ix!t=lj_ilj_old(q)
ENDIF

IF(lj_ulj_old(q).ge.0.0_sdk)THEN
   lj_rxb=lj_rholj(q)
   lj_ixb=lj_ilj_old(q)
ELSE
   lj_rxb=lj_rholj(q+(lj_ny+2))
   lj_ixb=lj_ilj_old(q+(lj_ny+2))
ENDIF

! lj_ryo=lj_rholj(q)
! lj_ryi=lj_rholj(q-1)
! lj_iyo=lj_ilj_old(q)
! lj_iyi=lj_ilj_old(q-1)
IF (j.eq.lj_nylj) THEN
! lj_ryo=lj_rholj(q)
! lj_iyo=lj_ilj_old(q)
ENDIF
! IF (j.eq.lj_nylj+1.or.j.eq.1) THEN
! lj_ryi=lj_rholj(q)
! lj_iyi=lj_ilj_old(q)
! ENDIF

IF (j.le.lj_nylj) THEN
   lj_ryo=lj_rholj(q)
   lj_iyo=lj_ilj_old(q)
   lj_ryi=lj_rholj(q-1)
   lj_iyi=lj_ilj_old(q-1)
ELSE
   lj_ryo=lj_rholj(q+1)
   lj_iyo=lj_ilj_old(q+1)
   lj_ryi=lj_rholj(q)
   lj_iyi=lj_ilj_old(q)
ENDIF

! General form of the continuity equation
!
lj_z(k6,k+k+k)=lj_drhodi(k)*lj_r(k)*lj_dx*lj_dy(k)+lj_r(k)*lj_dy(k)*lj_dt*(lj_rxb*lj_c2(k+lj_ny)*(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)-lj_rxt*lj_c2(k)*(1.0_sdk+lj_kidrag(k)*lj_dt))+lj_r(k)*lj_dx*lj_dt*lj_rholj(q)*(lj_d2(k+1+(i-1))-lj_d2(k+(i-1)))+0.5_sdk*(lj_d2(k+(i-1))+lj_d2(k+1+(i-1)))*((lj_ryo-lj_ryi)*lj_r(k)*lj_dx*lj_dt+lj_rholj(q)*lj_dx*lj_dy(k)*lj_dt)
(lj_c1(k)+lj_c0(k))*(1.0_sdk*lj_kidrag(k)*lj_dt)+lj_r(k) *
lj_dx*lj_dt*lj_rhoij(q)*(lj_d1(k+1)+(i-1))-lj_d0(k+(i-1)))+
(lj_d1(k+1)+(i-1)+lj_d0(k+1)+(i-1))+0.5_sdk*(lj_d0(k+(i-1))+
lj_d1(k+1)+(i-1)+lj_d0(k+1)+(i-1)))+
(lj_ryo-lj_ryi)*lj_r(k)*lj_dx*lj_dy(k)*lj_dt)

IF (j.eq.lj_nylj) THEN
lj_z(k6,k+k+k)=lj_z(k6,k+k+k)+lj_qil(k)/(lj_vol(k)*
(lj_ilj_old(q+1)-lj_if(k+1)))*lj_r(k)*lj_dy(k)*lj_dx*lj_dt
ENDIF

IF (j.eq.lj_nylj+1) THEN
lj_z(k1,k+k+1)=lj_z(k1,k+k+1)+lj_qil(k-1)/(lj_vol(k)*
(lj_ilj_old(q)-lj_if(k)))*lj_difdp(k)*lj_r(k)*lj_dy(k)*lj_dx*lj_dt
ENDIF

IF (j.ne.lj_ny) THEN
lj_z(k2,3_sik*(k+1)-1)=(lj_d0(k+1+(i-1))+lj_d1(k+1+(i-1)))+
(lj_r(k)*lj_rhoij(q)*lj_dx*lj_dt+0.5_sdk*(lj_ryo-lj_ryi)*
lj_r(k)*lj_dx*lj_dt+0.5_sdk*lj_rholj(q)*lj_dx*lj_dy(k)*lj_dt)
ENDIF

IF (j.eq.lj_nylj) THEN
lj_z(k2,3_sik*(k+1)-1)=lj_z(k2,3_sik*(k+1)-1)-lj_qil(k)/(lj_vol(k)*
(lj_ilj_old(q+1)-lj_if(k+1)))*lj_difdp(k+1)*lj_r(k)*lj_dy(k)*lj_dx*lj_dt
ENDIF

IF (j.eq.lj_nylj+1) THEN
lj_z(k8,3_sik*(k-1))=lj_d2(k+(i-1))+(lj_r(k)*lj_rhoij(q)*lj_dx*lj_dt)
ENDIF

IF (j.ne.1) THEN
lj_z(k3,3_sik*(k-1)-1)=(lj_d0(k+(i-1))-lj_d1(k+(i-1)))+
(lj_r(k)*lj_rhoij(q)*lj_dx*lj_dt-0.5_sdk*(lj_ryo-lj_ryi)*
lj_r(k)*lj_dx*lj_dt-0.5_sdk*lj_rholj(q)*lj_dx*lj_dy(k)*lj_dt)
ENDIF

IF (j.eq.lj_nylj+1) THEN
lj_z(k8,3_sik*(k-1))=lj_z(k8,3_sik*(k-1))-lj_qil(k-1)/(lj_vol(k)*
(lj_ilj_old(q)-lj_if(k)))*lj_difdp(k-1)*lj_r(k)*lj_dy(k)*lj_dx*lj_dt
ENDIF

IF (i.ne.lj_nx) THEN
lj_z(k4,3_sik*(k+lj_ny)-1)=lj_r(k)*lj_rxb*
(lj_c0(k+lj_ny)+lj_c1(k+lj_ny))*
(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)*lj_dy(k)*lj_dt
ENDIF

IF (i.ne.1) THEN
lj_z(k5,3_sik*(k-lj_ny)-1)=lj_r(k)*lj_rxt*(lj_c0(k)-lj_c1(k))*
(1.0_sdk+lj_kidrag(k)*lj_dt)*lj_dy(k)*lj_dt
ENDIF
lj_z(k10,3_sik*(k-lj_ny))=-1.0_sdk*lj_r(k)*lj_rxt*lj_c2(k)* &
     (1.0_sdk+lj_kidrag(k)*lj_dt)*lj_dy(k)*lj_dt
ENDIF

! For j=1, add the j-1 term to j
IF (j.eq.1) THEN
  lj_z(k1,k+k+k-1)=lj_z(k1,k+k+k-1)+(lj_d0(k+(i-1))-lj_d1(k+(i-1)))* &
    (lj_rholj(q)*lj_r(k)*lj_dx*lj_dt-0.5_sdk*(lj_ryo-lj_ryi)* &
     lj_r(k)*lj_dx*lj_dt-0.5_sdk*lj_rholj(q)*lj_dx*lj_dy(k)*lj_dt)
ENDIF
ENDDO

END SUBROUTINE TwoJetLHS

SUBROUTINE TwoJetRHS

!*****************************************************************************!
!     This subroutine Develops the right hand side of the equations for       !
!        the 2-D jet                                                         !
!                                                                             !
!     Programmed by Frank Buschman                                           !
!       rev. 4  7/24/2007                                                   !
!*****************************************************************************!

IMPLICIT NONE

INTEGER(sik) :: i,j,k,q

DO i=1,lj_nx
  DO j=1,lj_ny
    k=j+(i-1)*lj_ny
    q=(j+1)+i*(lj_ny+2)
!      IF (lj_vlj_old(q+1+i).gt.0.0_sdk)THEN
!         lj_ryo=lj_rholj(q)
!         lj_iyo=lj_ilj_old(q)
!      ELSE
!         lj_ryo=lj_rholj(q+1)
!         lj_iyo=lj_ilj_old(q+1)
!      ENDIF
!      IF(lj_ulj_old(q).ge.0.0_sdk)THEN
!         lj_rxt=lj_rholj(q-(lj_ny+2))
!         lj_ixt=lj_ilj_old(q-(lj_ny+2))
!      ELSE
!         lj_rxt=lj_rholj(q)
!         lj_ixt=lj_ilj_old(q)
!      ENDIF
    ! IF(lj_vlj_old(q+i).gt.0.0_sdk) THEN
    !    lj_ryo=lj_rholj(q)
    !    lj_iyo=lj_ilj_old(q)
    !  ELSE
    !    lj_ryo=lj_rholj(q+1)
    !    lj_iyo=lj_ilj_old(q+1)
    !  ENDIF
    ! IF(lj_vlj_old(q+i).gt.0.0_sdk) THEN
    !    lj_ryo=lj_rholj(q-1)
    !    lj_iyo=lj_ilj_old(q-1)
    !  ELSE
    !    lj_ryo=lj_rholj(q)
    !    lj_iyo=lj_ilj_old(q)
    !  ENDIF
    IF(lj_ulj_old(q).ge.0.0_sdk) THEN
      lj_rxt=lj_rholj(q-1)
      lj_ixt=lj_ilj_old(q-1)
    ELSE
      lj_rxt=lj_rholj(q)
      lj_ixt=lj_ilj_old(q)
    ENDIF
  ENDDO
END SUBROUTINE TwoJetRHS
lj_rxt=lj_rholj(q)
lj_ixt=lj_ilj_old(q)
ENDIF

IF(lj_ulj_old(q).ge.0.0_sdk)THEN
lj_rxb=lj_rholj(q)
lj_ixb=lj_ilj_old(q)
ELSE
lj_rxb=lj_rholj(q+(lj_ny+2))
lj_ixb=lj_ilj_old(q+(lj_ny+2))
ENDIF

lj_vy=0.5_sdk*(lj_vlj_old(q+i)+lj_vlj_old(q+1+i))
!
 lj_ilj_g=0.0_sdk
!
 lj_di=1j_ilj_old(q+(lj_ny+2))-2.0_sdk*1j_ilj_old(q)+
 !
1j_ilj_old(q-(lj_ny+2))
lj_di=0.0_sdk

lj_diy=1j_ilj_old(q+1)-2.0_sdk*1j_ilj_old(q)+1j_ilj_old(q-1)
lj_diy2=0.5_sdk*(1j_ilj_old(q+1)-1j_ilj_old(q-1))

IF (j.eq.lj_nylj) THEN
lj_diy=-(1j_ilj_old(q)-2.0_sdk*1j_ilj_old(q-1)+1j_ilj_old(q-2))
lj_diy2=(1j_ilj_old(q)-1j_ilj_old(q-1))
ENDIF

IF (j.eq.lj_nylj+1) THEN
lj_diy=1j_ilj_old(q)-2.0_sdk*1j_ilj_old(q+1)+1j_ilj_old(q+2)
lj_diy2=1j_ilj_old(q+1)-1j_ilj_old(q)
ENDIF

IF (j.eq.lj_nylj) THEN
lj_ilj_g(k)=lj_if(k+1)
ENDIF
IF (j.eq.lj_nylj+1) THEN
lj_ilj_g(k)=lj_ilj_old(q)
ENDIF

! lj_ryo=lj_rholj(q)
! lj_ryi=lj_rholj(q-1)
! lj_iyo=lj_ilj_old(q)
! lj_iyi=lj_ilj_old(q-1)
! IF (j.eq.lj_nylj) THEN
! lj_ryo=lj_rholj(q)
! lj_iyo=lj_ilj_old(q)
! ENDIF
! IF (j.eq.lj_nylj+1) THEN
! lj_ryi=lj_rholj(q)
! lj_iyi=lj_ilj_old(q)
! ENDIF

IF (j.le.lj_nylj) THEN
lj_ryo=lj_rholj(q)
lj_iyo=lj_ilj_old(q)
lj_ryi=lj_rholj(q-1)
lj_iyi=lj_ilj_old(q-1)
ELSE
lj_ryo=lj_rholj(q+1)
ENDIF
lj_iyo = lj_ilj_old(q+1)
lj_ryi = lj_rholj(q)
lj_iyi = lj_ilj_old(q)
ENDIF

! Right hand side general equation for the continuity equation
IF (j.eq.lj_nylj) THEN
  lj_b(k+k+k-1) = lj_qil(k) / (lj_vol(k) * (lj_ilj_old(q+1) - lj_if(k+1)))
  (lj_if(k+1) - lj_difdp(k+1) * lj_pres_old(q+1)) * lj_r(k) * lj_dx*
  lj_dy(k) * 1.0_dxt + lj_drhodi(k) * lj_r(k) * lj_ilj_old(q+1) * lj_dx*
  lj_dy(k) + 1.0_dxt + lj_drhodp(k) * lj_r(k) * lj_difdp(k+1) * lj_dy(k) +
  (lj_u0(k) * lj_rxt - lj_u0(k+lj_ny)) * lj_r(k) * lj_dx*lj_dy(k)*lj_dt +
  lj_dt + (lj_v0(k*(i+1)) - lj_v0(k+1*(i-1))) * lj_drholj(q)*lj_dy(k) *
  lj_r(k) * lj_dx*lj_dt - 0.5_sdk * ((lj_ryo-lj_ryi) * lj_r(k) * lj_dx*lj_dt +
  (lj_ryo-lj_ryi) * lj_r(k) * lj_dx*lj_dt + lj_drholj(q)*lj_dy(k)*lj_dt)
ELSEIF (j.eq.lj_nylj+1) THEN
  lj_b(k+k+k-1) = -lj_qil(k-1) / (lj_vol(k) * (lj_ilj_old(q) - lj_if(k)))
  (lj_if(k) - lj_difdp(k) * lj_pres_old(q)) * lj_r(k) * lj_dx*
  lj_dy(k) * 1.0_dxt + lj_drhodi(k) * lj_r(k) * lj_ilj_old(q) * lj_dx*
  lj_dy(k) + 1.0_dxt + lj_drhodp(k) * lj_r(k) * lj_difdp(k) * lj_dy(k) -
  lj_u0(k+lj_ny) * lj_rxb * (1.0_sdk + lj_kidrag(k+lj_ny) * lj_dt) *
  lj_r(k) * lj_dy(k) * lj_dt + (lj_rxb * lj_ulj_old(q*(i+1)) * lj_kidrag(k) *
  lj_dy(k) + lj_dt + (lj_v0(k*(i+1)) - lj_v0(k+1*(i-1))) * (lj_ryo-lj_ryi) *
  lj_r(k) * lj_dx * lj_dt + lj_drholj(q) * lj_dx * lj_dy(k) * lj_dt)
ELSE
  lj_b(k+k+k-1) = lj_drhodi(k) * lj_r(k) * lj_ilj_old(q) * lj_dx*lj_dy(k) +
  lj_drhodp(k) * lj_r(k) * lj_ilj_old(q) * lj_dx*lj_dy(k) +
  (lj_u0(k) * lj_rxt * (1.0_sdk + lj_kidrag(k)) * lj_dt) -
  lj_u0(k+lj_ny) * lj_rxb * (1.0_sdk + lj_kidrag(k+lj_ny) * lj_dt) *
  lj_r(k) * lj_dx*lj_dt + lj_drholj(q) * lj_dx*lj_dy(k) * lj_dt +
  lj_r(k) * lj_dx*lj_dy(k) * lj_dt + (lj_rxb * lj_ulj_old(q+1) -
  lj_u0(k-1)) * lj_kidrag(k+lj_ny) -
  lj_r(k) * lj_dt + (lj_v0(k*(i+1)) - lj_v0(k+1*(i-1))) * (lj_ryo-lj_ryi) *
  lj_r(k) * lj_dx*lj_dt + lj_drholj(q) * lj_dx*lj_dy(k) * lj_dt -
  0.5_sdk * (lj_v0(k*(i+1)) - lj_v0(k+1*(i-1))) * (lj_ryo-lj_ryi) *
  lj_r(k) * lj_dx*lj_dt + lj_drholj(q) * lj_dx*lj_dy(k) * lj_dt)
ENDIF

! Right hand side general equation for the energy equation
IF (j.eq.lj_nylj) THEN
  lj_b(k+k+k) = lj_ilj_old(q) * (lj_drholj(q) + lj_ilj_old(q)) *
  lj_difdp(k) * lj_pres_old(q) * lj_r(k) * lj_dx*lj_dy(k)*lj_dt +
  lj_qil(k) / lj_vol(k) * (1.0 + lj_if(k+1)) / (lj_ilj_old(q+1) -
  lj_if(k+1)) * (lj_if(k+1) - lj_difdp(k+1) * lj_pres_old(q+1)) * lj_r(k) *
  lj_dx*lj_dy(k) * lj_dt + (lj_ilj_old(q) * lj_drhodp(k) -
  lj_psft2btu) * lj_pres_old(q) * lj_r(k) * lj_dx*lj_dy(k) * lj_dt +
  (lj_if(k+1)) * lj_difdp(k+1) * lj_pres_old(q+1) * lj_r(k) * lj_dx*
  lj_dy(k) * lj_dx*lj_dy(k) * lj_dt + (lj_psft2btu) * lj_pres_old(q) *
  lj_r(k) * lj_dx*lj_dy(k) * lj_dt + (lj_if(k)) * lj_difdp(k) *
  lj_pres_old(q+1) * lj_r(k) * lj_dx*lj_dy(k) * lj_dt +
  (lj_ilj_old(q) * lj_drhodp(k-1) - lj_psft2btu) * lj_pres_old(q) *
  lj_r(k) * lj_dx*lj_dy(k) * lj_dt + (lj_psft2btu) * lj_pres_old(q) *
  lj_r(k) * lj_dx*lj_dy(k) * lj_dt + (lj_if(k)) * lj_difdp(k) *
  lj_pres_old(q+1) * lj_r(k) * lj_dx*lj_dy(k) * lj_dt +
  (lj_psft2btu) * lj_pres_old(q) * lj_r(k) * lj_dx*lj_dy(k) * lj_dt +
  (lj_if(k)) * lj_difdp(k) * lj_pres_old(q+1) * lj_r(k) * lj_dx*
  lj_dy(k) * lj_dx*lj_dy(k) * lj_dt + (lj_psft2btu) * lj_pres_old(q) *
  lj_r(k) * lj_dx*lj_dy(k) * lj_dt + (lj_if(k)) * lj_difdp(k) *
  lj_pres_old(q+1) * lj_r(k) * lj_dx*lj_dy(k) * lj_dt +
  (lj_psft2btu) * lj_pres_old(q) * lj_r(k) * lj_dx*lj_dy(k) * lj_dt +
  (lj_if(k)) * lj_difdp(k) * lj_pres_old(q+1) * lj_r(k) * lj_dx*
  lj_dy(k) * lj_dx*lj_dy(k) * lj_dt + (lj_psft2btu) * lj_pres_old(q) *
  lj_r(k) * lj_dx*lj_dy(k) * lj_dt + (lj_if(k)) * lj_difdp(k) *
  lj_pres_old(q+1) * lj_r(k) * lj_dx*lj_dy(k) * lj_dt +
  (lj_psft2btu) * lj_pres_old(q) * lj_r(k) * lj_dx*lj_dy(k) * lj_dt +
  (lj_if(k)) * lj_difdp(k) * lj_pres_old(q+1) * lj_r(k) * lj_dx*lj dy(k) * lj_dt -
  0.5_sdk * (lj_v0(k*(i+1)) - lj_v0(k+1*(i-1))) * (lj_ryo-lj_ryi) *
  lj_r(k) * lj_dx*lj_dy(k) * lj_dt)
ELSE
  lj_b(k+k+k) = lj_drhodi(k) * lj_r(k) * lj_ilj_old(q) * lj_dx*lj_dy(k) +
  lj_drhodp(k) * lj_r(k) * lj_ilj_old(q) * lj_dx*lj_dy(k) +
  (lj_u0(k) * lj_rxt * (1.0_sdk + lj_kidrag(k)) * lj_dt) -
  lj_u0(k+lj_ny) * lj_rxb * (1.0_sdk + lj_kidrag(k+lj_ny) * lj_dt) *
  lj_r(k) * lj_dx*lj_dt + lj_drholj(q) * lj_dx*lj_dy(k) * lj_dt +
  (lj_v0(k*(i+1)) - lj_v0(k+1*(i-1))) * (lj_ryo-lj_ryi) *
  lj_r(k) * lj_dx*lj_dt + lj_drholj(q) * lj_dx*lj_dy(k) * lj_dt)
ENDIF
ELSEIF (j.eq.lj_nylj+1) THEN

 lj_b(k+k+k)=lj_ilj_old(q)*(lj_rholj(q)+lj_ilj_old(q)*lj_drhodi(k))* &
 lj_r(k)*lj_dx*lj_dy(k)*lj_dy(k)+lj_ilj_old(q)*lj_drhodp(k)* &
 lj_pres_old(q)*lj_r(k)*lj_dx*lj_dy(k)*lj_dy(k)+ &
 lj_r(k)*(lj_u0(k)*lj_rxt*lj_ixt*(1.0_sdk+lj_kidrag(k)*lj_dt)- &
 lj_u0(k+lj_ny)*lj_rxb*lj_ixb*(1.0_sdk+lj_kidrag(k+lj_ny)))* &
 lj_dxi1*lj_dy(k)+lj_dxi1*lj_dy(k)*lj_dyi1+lj_dx*lj_dy(k)*lj_dyi2 &
lj_b(k+k+k)=lj_ilj_old(q)*(lj_rholj(q)+lj_ilj_old(q)*lj_drhodi(k))* &
 lj_r(k)*lj_dx*lj_dy(k)*lj_dy(k)+lj_ilj_old(q)*lj_drhodp(k)* &
 lj_pres_old(q)*lj_r(k)*lj_dx*lj_dy(k)*lj_dy(k)+ &
 lj_r(k)*(lj_u0(k)*lj_rxt*lj_ixt*(1.0_sdk+lj_kidrag(k)*lj_dt)- &
 lj_u0(k+lj_ny)*lj_rxb*lj_ixb*(1.0_sdk+lj_kidrag(k+lj_ny)))* &
 lj_dxi1*lj_dy(k)+lj_dxi1*lj_dy(k)*lj_dyi1+lj_dx*lj_dy(k)*lj_dyi2 &
lj_b(k+k+k-1)=lj_b(k+k+k-1)-(lj_r(k)*lj_rxt*(lj_c0(k)-lj_c1(k))* &
(1.0_sdk+lj_kidrag(k)*lj_dt)*lj_dy(k)*lj_dt)*lj_pres_old(q-(lj_ny+2))+(1.0_sdk*lj_r(k)*lj_rxt*lj_c2(k)* &
(1.0_sdk+lj_kidrag(k)*lj_dt)*lj_ixt*lj_dx*lj_dy(k)*lj_dy(k)* &
lj_dt)*lj_ilj_old(q-(lj_ny+2))
ENDIF

! At i=1, the i-1 term from the LHS is moved to the RHS
IF (i.eq.1) THEN
 lj_b(k+k+k-1)=lj_b(k+k+k-1)-(lj_r(k)*lj_rxt*(lj_c0(k)-lj_c1(k))* &
(1.0_sdk+lj_kidrag(k)*lj_dt)*lj_dy(k)*lj_dt)*lj_pres_old(q-(lj_ny+2))+(1.0_sdk*lj_r(k)*lj_rxt*lj_c2(k)* &
(1.0_sdk+lj_kidrag(k)*lj_dt)*lj_dy(k)*lj_dt)*lj_ilj_old(q-(lj_ny+2))
ENDIF
At i=nx, the i+1 term from the LHS is moved to the RHS

IF (i.eq.lj_nx) THEN

lj_b(k+k+k-1)=lj_b(k+k+k-1)-(lj_r(k)*lj_rxb*(lj_c0(k+lj_ny)+
lj_c1(k+lj_ny))*(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)*
lj_dy(k)*lj_dt)*lj_pres_old(q+(lj_ny+2))-(lj_r(k)*lj_rxb*lj_c2(k+lj_ny)*
(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)*lj_dy(k)*lj_dt)*lj_ilj_old(q+(lj_ny+2))

ENDIF

At j=ny, the j+1 term from the LHS is moved to the RHS

IF (j.eq.lj_ny) THEN

lj_b(k+k+k)=lj_b(k+k+k)-(lj_r(k)*lj_rxb*(lj_c0(k+lj_ny)+
lj_c1(k+lj_ny))*(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)*
lj_ixb*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt)*lj_pres_old(q+(lj_ny+2))-(lj_r(k)*lj_rxb*lj_c2(k+lj_ny)*
(1.0_sdk+lj_kidrag(k+lj_ny)*lj_dt)*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt)*lj_ilj_old(q+(lj_ny+2))

ENDIF

lj_b(k+k+k)=lj_ilj_old(q+1)

END DO

lj_b_old=lj_b

END SUBROUTINE TwoJetRHS

SUBROUTINE JetAxMom

!*****************************************************************************!
!     This subroutine linearizes the axial momentum equation                  !
!               for the 2-D jet for use in                                    !
!         the energy and contiuity equations                                 !
!*****************************************************************************!

DO k=1,lj_k
  IF(lj_b_old(k).ne.0.0)THEN
    lj_berr=(lj_b(k)-lj_b_old(k))/lj_b_old(k)*100
  ENDIF
ENDDO

END SUBROUTINE JetAxMom
IMPLICIT NONE

INTEGER(sik) :: i,j,k,q
REAL(sdk) :: gu,rb,rt,ro,ri,ru,rad,vo,vi,uo,ui,ut,ub,utf,ubf,ug
REAL(sdk) :: lj_drdp,lj_drdi,iu,pu,vu,dux,duy,duy2

DO i=1,lj_nx+1
  DO j=1,lj_ny
    k=j+(i-1)*lj_ny
    q=(j+1)+i*(lj_ny+2)
   !
    ! Set terms for use in the axial momentum equation
    !
    IF(i.eq.1)THEN
      ! mass transfer coefficient
      gu=0.5_sdk*(lj_glj(q)/lj_vol(k)+lj_glj(q-(lj_ny+2))/lj_vol(k))
    ELSEIF(i.eq.lj_nx+1)THEN
      gu=0.5_sdk*(lj_glj(q)/lj_vol(k-lj_ny)+lj_glj(q-(lj_ny+2))/lj_vol(k-lj_ny))
    ELSE
      gu=0.5_sdk*(lj_glj(q)/lj_vol(k)+lj_glj(q-(lj_ny+2))/lj_vol(k-lj_ny))
    ENDIF

    IF(j.eq.lj_nylj)THEN
      gu=-gu
    ENDIF
    gu=0.0_sdk
    IF(j.eq.lj_ny+lj_nylj)THEN
      gu=lj_ulj_old(q)
    ELSEIF(j.eq.lj_nylj)THEN
      ug=lj_ulj_old(q+1)
    ENDIF
    ug=0.0
    ug=0.0
    
    radial velocity at outside face
    vo=0.5_sdk*(lj_vlj_old(q+1)+lj_vlj_old(q+(i-1)-(lj_ny+2)))
    
    radial velocity at inside face
    vi=0.5_sdk*(lj_vlj_old(q+i)+lj_vlj_old(q+(i-1)-(lj_ny+2)))
    
    radial velocity in the momentum cell
    vu=0.5_sdk*(vo+vi)
    
    axial velocity at top face
    ut=0.5_sdk*(lj_ulj_old(q)+lj_ulj_old(q-(lj_ny+2)))
    
    axial velocity at bottom face
    ub=0.5_sdk*(lj_ulj_old(q)+lj_ulj_old(q+(lj_ny+2)))
    
    density fluxed at the bottom face
    IF (ub.ge.0.0_sdk) THEN
      rb=0.5_sdk*(lj_rholj(q)+lj_rholj(q-(lj_ny+2)))
    ELSE
      IF (i.ne.lj_nx+1)THEN
        rb=0.5_sdk*(lj_rholj(q)+lj_rholj(q+(lj_ny+2)))
      ENDIF
ELSE
    rb=lj_rholj(q)
ENDIF
ENDIF

! density fluxed at the top face
IF (ut.ge.0.0_sdk) THEN
    IF(i.ne.1) THEN
        rt=0.5_sdk*(lj_rholj(q-(lj_ny+2))+lj_rholj(q-(lj_ny+2)-(lj_ny+2)))
    ELSE
        rt=lj_rholj(q-(lj_ny+2))
    ENDIF
ELSE
    rt=0.5_sdk*(lj_rholj(q)+lj_rholj(q-(lj_ny+2)))
ENDIF

! density within the cell
ru=0.5_sdk*(lj_rholj(q)+lj_rholj(q-(lj_ny+2)))
IF(j.le.lj_nylj) THEN
    ro=0.5_sdk*(lj_rholj(q)+lj_rholj(q-(lj_ny+2)))
    ri=0.5_sdk*(lj_rholj(q-1)+lj_rholj(q-1-(lj_ny+2)))
ELSE
    ro=0.5_sdk*(lj_rholj(q+1)+lj_rholj(q+1-(lj_ny+2)))
    ri=0.5_sdk*(lj_rholj(q)+lj_rholj(q-(lj_ny+2)))
ENDIF

! radius of the momentum cell
IF(i.ne.1.and.i.eq.lj_nx+1) THEN
    rad=0.5_sdk*(lj_r(k)+lj_r(k-lj_ny))
ELSEIF(i.eq.lj_nx+1) THEN
    rad=lj_r(k-lj_ny)
ELSE
    rad=lj_r(k)
ENDIF

! axial velocity fluxed at top face
IF (ut.ge.0.0_sdk) THEN
    utf=lj_ulj_old(q-(lj_ny+2))
ELSE
    utf=lj_ulj_old(q)
ENDIF

! axial velocity fluxed at bottom face
IF(ub.ge.0.0_sdk) THEN
    ubf=lj_ulj_old(q)
ELSE
    ubf=lj_ulj_old(q+(lj_ny+2))
ENDIF

! axial velocity fluxed at outside face
IF(vo.gt.0.0_sdk) THEN
    uo=lj_ulj_old(q)
ELSE
    uo=lj_ulj_old(q+1)
ENDIF

! axial velocity fluxed at inside face
IF(vi.gt.0.0_sdk) THEN
    ui=lj_ulj_old(q-1)
ELSE

ui = lj_ulj_old(q)
ENDIF

! enthalpy in momentum cell
iu = 0.5 *sdk * (lj_ilj_old(q) + lj_ilj_old(q - (lj_ny + 2)))

! pressure in momentum cell
pu = 0.5 *sdk * (lj_pres_old(q) + lj_pres_old(q - (lj_ny + 2)))

IF (i .ne. 1 .and. i .ne. lj_nx + 1) THEN
  lj_drdp = 0.5 *sdk * (lj_drhodp(k) + lj_drhodp(k - lj_ny))
  lj_drdi = 0.5 *sdk * (lj_drhodi(k) + lj_drhodi(k - lj_ny))
ELSEIF (i .eq. lj_nx + 1) THEN
  lj_drdp = lj_drhodp(k - lj_ny)
  lj_drdi = lj_drhodi(k - lj_ny)
ELSE
  lj_drdp = lj_drhodp(k)
  lj_drdi = lj_drhodi(k)
ENDIF

! dux = (lj_ulj_old(q + (lj_ny + 2)) - 2.0 *sdk * lj_ulj_old(q) +
  lj_ulj_old(q - (lj_ny + 2))) / (lj_dx * lj_dx)
dux = 0.0 *sdk

IF (j .eq. lj_nylj) THEN
  duy = -(lj_ulj_old(q) - 2.0 *sdk * lj_ulj_old(q - 1) +
  lj_ulj_old(q - 2)) / (lj_dyx(k) * lj_dyx(k))
  duy2 = -(lj_ulj_old(q) - lj_ulj_old(q - 1)) / (rad * lj_dyx(k))
ELSEIF (j .eq. lj_nylj + 1) THEN
  duy = (lj_ulj_old(q + 2) - 2.0 *sdk * lj_ulj_old(q + 1) +
  lj_ulj_old(q)) / (lj_dyx(k) * lj_dyx(k))
  duy2 = (lj_ulj_old(q + 1) - lj_ulj_old(q)) / (rad * lj_dyx(k))
ELSE
  duy = (lj_ulj_old(q + 1) - 2.0 *sdk * lj_ulj_old(q) +
  lj_ulj_old(q - 1)) / (lj_dyx(k) * lj_dyx(k))
  duy2 = (lj_ulj_old(q + 1) - lj_ulj_old(q - 1)) / (2.0 *sdk * rad * lj_dyx(k))
ENDIF

lj_u0(k) = lj_ulj_old(q) - lj_dt * (rb * ub * ubf - rt * ut * utf) / (ru * lj_dx) -
  lj_dt * ((lj_ulj_old(q) * (vo - vi) + vu * (uo - ui)) +
  lj_ulj_old(q) * vu * (ro - ri) / ru) / lj_dyx(k) +
  lj_ulj_old(q) * vu / rad +
  lj_ulj_old(q) * vu / rad) + gu * ug * lj_dt / ru +
  lj_edmx(k) * lj_dt * (dux + duy + duy2)

lj_c0(k) = -1.0 *sdk * lj_gc * lj_sft2sin * lj_dt / (ru * lj_dx)
lj_c1(k) = -0.5 *sdk * lj_ulj_old(q) * lj_drdp / ru
lj_c2(k) = -0.5 *sdk * lj_ulj_old(q) * lj_drdi / ru

IF (j .lt. lj_nylj + 1 .and. i .eq. 1) THEN
  lj_u0(k) = lj_uin
  lj_c0(k) = 0.0 * sdk
  lj_c1(k) = 0.0 * sdk
  lj_c2(k) = 0.0 * sdk
ENDIF

! IF (j .gt. lj_nylj .and. i .eq. 1) THEN
!  lj_u0(k) = 0.0 * sdk
!  lj_c0(k) = 0.0 * sdk
!  lj_c1(k) = 0.0 * sdk
!  lj_c2(k) = 0.0 * sdk
! ENDIF
SUBROUTINE JetRadMom

!*****************************************************************************!
!     This subroutine linearizes the radial momentum equation for           !
!         the 2-D jet for use in the energy and continuity equations         !
!                                                                             !
!     Programmed by Frank Buschman                                            !
!       rev. 0 7/24/2007                                                      !
!*****************************************************************************!

IMPLICIT NONE

INTEGER(sik) :: i,j,k,q
REAL(sdk) :: gv,rb,rt,ro,ri,rv,vo,vi,vt,vb,ut,ub,vof,vif,vg
REAL(sdk) :: lj_drdp,lj_drdi,iv,pv,dvx,dvy,dvy2

DO i=1,lj_nx
   DO j=1,lj_ny+1
      k=j+(i-1)*(lj_ny+1)
      q=(j+1)+i*(lj_ny+3)

      IF(j.eq.lj_ny+1)THEN
         gv=0.5_sdk*(lj_glj(q-i)/lj_vol(k-1-(i-1))+                           &
          lj_glj(q-1-i)/lj_vol(k-1-(i-1)))
      ELSEIF(j.eq.1)THEN
         gv=0.5_sdk*(lj_glj(q-i)/lj_vol(k-(i-1))+lj_glj(q-1-i)/lj_vol(k-(i-1)))
      ELSE
         gv=0.5_sdk*(lj_glj(q-i)/lj_vol(k-(i-1))+                             &
          lj_glj(q-1-i)/lj_vol(k-1-(i-1)))
      ENDIF

      !     gv=0.0_sdk
      rv=0.5_sdk*(lj_rholj(q-i)+lj_rholj(q-1-i))
      !     velocity at the outide face
      vo=lj_vlj_old(q+1)
      !     velocity at the inside face
      vi=lj_vlj_old(q)
      !     axial velocity at the top face
      ut=0.5_sdk*(lj_ulj_old(q-i)+lj_ulj_old(q-1-i))
      !     axial velocity at the bottom face
      ub=0.5_sdk*(lj_ulj_old(q+(lj_ny+3)-(i+1))+                              &
          lj_ulj_old(q-1+(lj_ny+3)-(i+1)))
      !     velocity fluxed at the top
      IF(ut.ge.0.0_sdk)THEN
         vt=lj_vlj_old(q-(lj_ny+3))
      ELSE
         vt=(-1.0_sdk)*lj_vlj_old(q-(lj_ny+3))
      ENDIF

   END DO

END DO

END SUBROUTINE JetRadMom
ELSE
   vt=lj_vlj_old(q)
ENDIF

! velocity fluxed at the bottom face
IF(ub.ge.0.0_sdk)THEN
   vb=lj_vlj_old(q)
ELSE
   vb=lj_vlj_old(q+(lj_ny+3))
ENDIF

! velocity fluxed at the outside face
IF(vo.gt.0.0_sdk)THEN
   vof=lj_vlj_old(q)
ELSE
   vof=lj_vlj_old(q+1)
ENDIF

! velocity fluxed at the inside face
IF(vi.gt.0.0_sdk)THEN
   vif=lj_vlj_old(q-1)
ELSE
   vif=lj_vlj_old(q)
ENDIF

IF(j.eq.lj_nylj)THEN
   !
   vg=lj_vlj_old(q+1)
   vg=0.0_sdk
ELSEIF(j.eq.lj_nylj+1)THEN
   vg=lj_vlj_old(q)
ENDIF

vg=0.0_sdk

! density fluxed at the top face
IF(ut.ge.0.0_sdk)THEN
   rt=0.5_sdk*(lj_rholj(q-(lj_ny+3)-(i-1))+lj_rholj(q-1-(lj_ny+3)-(i-1)))+
   lj_rholj(q-1-(lj_ny+3)-(i-1))
ELSE
   rt=rv
ENDIF

! density fluxed at the bottom face
IF(ub.ge.0.0_sdk)THEN
   rb=rv
ELSE
   rb=0.5_sdk*(lj_rholj(q+(lj_ny+3)-(i+1))+lj_rholj(q-1+(lj_ny+3)-(i+1)))+
   lj_rholj(q-1+(lj_ny+3)-(i+1))
ENDIF

! density fluxed at the outside face
IF (j.le.lj_nylj+1) THEN
   ro=lj_rholj(q-i)
   ri=lj_rholj(q-1-i)
   iv=0.5_sdk*(lj_ilj_old(q-i)+lj_ilj_old(q-1-i))
   pv=0.5_sdk*(lj_pres_old(q-i)+lj_pres_old(q-1-i))
ENDIF

IF(j.ne.1.and.j.ne.lj_nylj+1)THEN
   lj_drhodp=0.5_sdk*(lj_drhodp(k-(i-1))+lj_drhodp(k-1-(i-1)))+
   lj_drhodp(k-1-(i-1))
   lj_drhodi=0.5_sdk*(lj_drhodi(k-(i-1))+lj_drhodi(k-1-(i-1)))+
   lj_drhodp(k-1-(i-1))
ELSEIF(j.eq.1)THEN
   lj_drhodp=0.5_sdk*(lj_drhodp(k-(i-1)))+
   lj_drhodp(k-1-(i-1))
lj_drdi=lj_drhodi(k-(i-1))
ELSE
lj_drdp=lj_drhodp(k-1-(i-1))
ENDIF

! dvx=(lj_vlj_old(q+(lj_ny+3))-2.0_sdk*lj_vlj_old(q)+                     
!     lj_vlj_old(q-(lj_ny+3)))/(lj_dx*lj_dx)
dvx=0.0_sdk

dvy=(lj_vlj_old(q+1)-2.0_sdk*lj_vlj_old(q)+                             
    lj_vlj_old(q-1))/(lj_dyy(k)*lj_dyy(k))
dvy2=(lj_vlj_old(q+1)-lj_vlj_old(q-1))/(2.0_sdk*lj_ry(k)*lj_dyy(k))
lj_edmy=0.0

IF (j.eq.1.or.j.eq.lj_nylj+1) THEN
! IF (j.le.lj_nylj+1) THEN
lj_v0(k)=0.0_sdk
lj_d0(k)=0.0_sdk
lj_d1(k)=0.0_sdk
lj_d2(k)=0.0_sdk
! ELSE IF (j.eq.lj_ny+1) THEN
! lj_v0(k)=lj_vi(i)
! lj_d0(k)=0.0_sdk
! lj_d1(k)=0.0_sdk
! lj_d2(k)=0.0_sdk
ELSE
lj_v0(k)=lj_vlj_old(q)-lj_dt*(rb*vb*ub-rt*vt*ut)/(rv*lj_dx)-                     
    lj_dt*{(2.0_sdk*lj_vlj_old(q)*(vo-vi)+lj_vlj_old(q)*         
     (ro-ri)/rv)/lj_dyy(k)+lj_vlj_old(q)*         
     (iv*lj_drdi/rv+pv*lj_drdp/rv)+lj_edmy(k)*         
     lj_dt*(dvx+dvy+dvy2))
lj_d0(k)=-1.0_sdk*lj_gc*lj_sft2sin*lj_dt/(rv*lj_dyy(k))
lj_d1(k)=-0.5_sdk*lj_vlj_old(q)*lj_drdp/rv
lj_d2(k)=-0.5_sdk*lj_vlj_old(q)*lj_drdi/rv
ENDIF
ENDDO
ENDDO

! lj_v0=0.0_sdk
! lj_d0=0.0_sdk
! lj_d1=0.0_sdk
! lj_d2=0.0_sdk
END SUBROUTINE JetRadMom

SUBROUTINE JetNonCon

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Subroutine JetNonCon sets the equation for the transport                 !
! of noncondensable gas in the vapor boundary layer                    !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

IMPLICIT NONE

INTEGER(sik) :: i, j, k, l, q, k1, k2, k3, k4, k5, k6, k7, k8, k9, k10, k11
REAL(sdk) :: ri, ro, rt, rb, gi, go, gt, gb, dgx, dgy, dgyl

END
! Set the left hand side of the non-con equation

DO i=1,lj_nx
  DO j=1,lj_nylj
    k=j+(i-1)*lj_ny
    IF (band) THEN
      k11=0
    ELSE
      k11=k+k+k-2
    ENDIF
    lj_z(k11,k+k+k-2)=1.0_sdk
    lj_b(k+k+k-2)=0.0_sdk
  ENDDO
  DO j=lj_nylj+1,lj_ny
    k=j+(i-1)*lj_ny
    q=j+1+i*(lj_ny+2)
    IF (band) THEN
      k1=lj_w ! p(k)         p(i,j)
      k2=lj_w+2 ! p(k+1)       p(i,j+1)
      k3=lj_w-2 ! p(k-2)       p(i,j-1)
      k4=lj_w+2*lj_ny ! p(k+lj_ny) p(i+1,j)
      k5=lj_w-2*lj_ny ! p(k-lj_ny) p(i-1,j)
      k6=lj_w+1 ! i(k)         i(i,j)
      k7=lj_w+3 ! i(k+1)       i(i,j+1)
      k8=lj_w-1 ! i(k-1)       i(i,j-1)
      k9=lj_w+2*lj_ny+1 ! i(k+lj_ny) i(i+1,j)
      k10=lj_w-2*lj_ny+1 ! i(k-lj_ny) i(i+1,j)
      k11=0.0 ! gas(k)       gas(i,j)
    ELSE
      k1=k+k+k-2 ! p(k)         p(i,j)
      k2=k+k+k-2 ! p(k+1)       p(i,j+1)
      k3=k+k+k-2 ! p(k-2)       p(i,j-1)
      k4=k+k+k-2 ! p(k+lj_ny) p(i+1,j)
      k5=k+k+k-2 ! p(k-lj_ny) p(i-1,j)
      k6=k+k+k-2 ! i(k)         i(i,j)
      k7=k+k+k-2 ! i(k+1)       i(i,j+1)
      k8=k+k+k-2 ! i(k-1)       i(i,j-1)
      k9=k+k+k-2 ! i(k+lj_ny) i(i+1,j)
      k10=k+k+k-2 ! i(k-lj_ny) i(i+1,j)
      k11=k+k+k-2 ! gas(k)       gas(i,j)
    ENDIF
    IF(lj_vlj_old(q+1+i).gt.0.0_sdk)THEN
      ro=lj_rholj(q)
      go=lj_gas_old(q)
    ELSE
      ro=lj_rholj(q+1)
      go=lj_gas_old(q+1)
    ENDIF
    IF(lj_vlj_old(q+i).gt.0.0_sdk)THEN
      ri=lj_rholj(q-1)
      gi=lj_gas_old(q-1)
    ELSE
      ri=lj_rholj(q)
    ENDIF
  ENDDO
ENDDO
gi = lj_gas_old(q)
ENDIF

IF(j.eq.lj_ny+lj+1)THEN
  ri = lj_rholj(q)
  gi = lj_gas_old(q)
ENDIF

IF(lj_ulj_old(q+(lj_ny+2)).ge.0.0_sdk)THEN
  rb = lj_rholj(q)
  gb = lj_gas_old(q+(lj_ny+2))
ELSE
  rb = lj_rholj(q+lj_ny+2)
  gb = lj_gas_old(lj+(lj_ny+2))
ENDIF

IF(lj_ulj_old(q).ge.0.0_sdk)THEN
  rt = lj_rholj(q-lj_ny)
  gt = lj_gas_old(q-lj_ny)
ELSE
  rt = lj_rholj(q)
  gt = lj_gas_old(q)
ENDIF

! coefficient for noncondensable gas at i,j
lj_z(k11,k+k+k-2) = lj_r(k)*lj_rholj(q)*lj_dx*lj_dx*lj_dy(k)*lj_dy(k)

lj_z(k6,k+k+k-1) = lj_r(k)*lj_gas_old(q)*lj_drhodp(k)*lj_dx*lj_dx*lj_dy(k)

! coefficient at j+1
IF(j.ne.lj_ny)THEN
lj_z(k2,3_sik*(k+1)-1) = lj_z(k2,3_sik*(k+1))
ENDIF

lj_z(k7,3_sik*(k+1)) = lj_z(k7,3_sik*(k+1))
! coefficients at j-1
IF (j.ne.lj_ny+1) THEN
  lj_z(k3,3_sik*(k-1)-1)=(lj_d0(k+(i-1))-lj_d1(k+(i-1)))*
    (lj_rholj(q)*lj_gas_old(q)-0.5_sdk*lj_rholj(q)*
     (go-gi)-0.5_sdk*lj_gas_old(q)*(ro-ri))*lj_r(k) &
    lj_dx*lj_dx*lj_dy(k)*lj_dt-0.5_sdk*lj_rholj(q)*
    lj_gas_old(q)*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt)
ENDIF

lj_z(k8,3_sik*(k-1))=lj_d2(k+(i-1))*((0.5_sdk*lj_rholj(q)*
  (go-gi)+0.5_sdk*lj_gas_old(q)*lj_rholj(q)*
  lj_gas_old(q))*lj_r(k)*lj_dx*lj_dx*lj_dy(k)*lj_dt+0.5_sdk*lj_rholj(q)*lj_gas_old(q)*lj_dx*  
  lj_dx*lj_dy(k)*lj_dy(k)*lj_dt)
ENDIF

! coefficients at i+1
IF (i.ne.lj_nx) THEN
  lj_z(k4,3_sik*(k+lj_ny)-1)=(lj_c0(k+lj_ny)+lj_c1(k+lj_ny))*lj_r(k)*
    rb*gb*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt
  lj_z(k9,3_sik*(k+lj_ny))=lj_c2(k+lj_ny)*lj_r(k)*rb*gb*lj_dx*  
    lj_dy(k)*lj_dy(k)*lj_dt
ENDIF

! coefficients at i-1
IF (i.ne.1) THEN
  lj_z(k5,3_sik*(k-lj_ny)-1)=(lj_c0(k)-lj_c1(k))*lj_r(k)*
    rt*gt*lj_dx*lj_dy(k)*lj_dy(k)*lj_dt
  lj_z(k10,3_sik*(k-lj_ny))=-1.0_sdk*lj_c2(k)*lj_r(k)*rt*gt*  
    lj_dx*lj_dy(k)*lj_dy(k)*lj_dt
ENDIF

ENDO
ENDDO

! Set the right hand side of the non-con equation
DO i=1,lj_nx
  DO j=lj_nylj+1,lj_ny
    k=j+(i-1)*lj_ny
    q=j+i*(lj_ny+2)
    IF(lj_vlj_old(q+i+1).gt.0.0_sdk) THEN
      ro=lj_rholj(q)
      go=lj_gas_old(q)
    ELSE
      ro=lj_rholj(q+1)
      go=lj_gas_old(q+1)
    ENDIF
    IF(lj_vlj_old(q+i).gt.0.0_sdk) THEN
      ri=lj_rholj(q-1)
      gi=lj_gas_old(q-1)
    ELSE
      ri=lj_rholj(q)
      gi=lj_gas_old(q)
    ENDIF
    IF(lj_ulj_old(q+lj_ny+2)).ge.0.0_sdk) THEN
      END
\[ rb = lj_rholj(q) \]
\[ gb = lj_gas_old(q) \]

**ELSE**
\[ rb = lj_rholj(q + (lj_ny + 2)) \]
\[ gb = lj_gas_old(q + (lj_ny + 2)) \]

**ENDIF**

**IF** \((lj_ulj_old(q) \geq 0.0\_sdk)\)**
\[ rt = lj_rholj(q - lj_ny) \]
\[ gt = lj_gas_old(q - lj_ny) \]

**ELSE**
\[ rt = lj_rholj(q) \]
\[ gt = lj_gas_old(q) \]

**ENDIF**

**IF** \((j \equiv lj_nylj + 1)\)**
\[ ri = lj_rholj(q) \]
\[ gi = lj_gas_old(q) \]
\[ dgy = lj_gas_old(q + 2) - 2.0\_sdk * lj_gas_old(q + 1) + lj_gas_old(q) \]
\[ dgy1 = lj_gas_old(q + 1) - lj_gas_old(q) \]

**ELSE**
\[ dgy = lj_gas_old(q - 1) - 2.0\_sdk * lj_gas_old(q) + lj_gas_old(q + 1) \]
\[ dgy1 = 0.5\_sdk * (lj_gas_old(q + 1) - lj_gas_old(q - 1)) \]

**ENDIF**

\[ dgx = lj_gas_old(q - lj_ny) - 2.0\_sdk * lj_gas_old(q + lj_gas_old(q + lj_ny)) \]

\[ lj_{b(k+k+k-2)} = lj_r(k) * lj_gas_old(q) * lj_dx * lj_dy(k) + \]
\[ lj_dy(k) * (lj_rholj(q) + lj_pres_old(q) * lj_dhdpd(k) + \]
\[ lj_ilj_old(q) * lj_dhodi(k)) + lj_edg(k) * lj_dt * (lj_r(k) + \]
\[ lj_dy(k) * lj_dy(k) + lj_r(k) * lj_dx * lj_dgdy + \]
\[ lj_dx * lj_dx * lj_1dy(dqy)) + lj_r(k) * lj_dx * lj_dy(k) + \]
\[ lj_dy(k) * lj_dy(k) + lj_u0(k) * rt * qt - lj_u0(k + lj_ny) * rb * gb + \]
\[ lj_r(k) * lj_dx * lj_dy(k) * lj_dt * lj_rholj(q) + \]
\[ lj_gas_old(q) * (lj_v0(k + (i-1)) - lj_v0(k + (i-1))) - \]
\[ 0.5\_sdk * (lj_v0(k + (i-1)) + lj_v0(k + (i-1))) * \]
\[ (lj_rholj(q) * (go - gi) + lj_gas_old(q) * (ro - ri)) + \]
\[ lj_r(k) * lj_dx * lj_dy(k) + lj_dy(k) * lj_dt * lj_rholj(q) + \]
\[ lj_gas_old(q) * lj_dx * lj_dy(k) * lj_dy(k) * lj_dt (l) \]

**IF** \((j \equiv lj_ny)\)**
\[ lj_{b(k+k+k-2)} = lj_{b(k+k+k-2)} + (lj_c0(k) + (lj_ny - 1)) + lj_{d1(k) + (lj_ny + 1)) * \]
\[ (lj_rholj(q) * lj_gas_old(q) + 0.5\_sdk * lj_rholj(q) * (go - gi) + \]
\[ 0.5\_sdk * lj_gas_old(q) * (ro - ri)) * lj_r(k) * lj_dx * lj_dy * \]
\[ lj_dy(k) * lj_dt + 0.5\_sdk * lj_rholj(q) * lj_gas_old(q) * lj_dx * \]
\[ lj_dy(k) * lj_dy(k) * lj_dt) * lj_pres_old(q + 1) + \]
\[ (lj_d2(k + (i-1)) + (lj_rholj(q) * lj_gas_old(q) + \]
\[ 0.5\_sdk * lj_rholj(q) * (go - gi) + 0.5\_sdk * lj_gas_old(q) * \]
\[ (ro - ri)) * lj_r(k) * lj_dx * lj_dy(k) * lj_dt + \]
\[ 0.5\_sdk * lj_rholj(q) * lj_gas_old(q) * lj_dx * lj_dy* \]
\[ lj_dy(k) * lj_dy(k) * lj_dt) * lj_ilj_old(q + 1) \]

**ENDIF**

**IF** \((i \equiv 1)\)**
\[ lj_{b(k+k+k-2)} = lj_{b(k+k+k-2)} + (lj_c0(k) - lj_cl(k)) * lj_r(k) * rt * qt + \]
\[ lj_dx * lj_dy(k) * lj_dy(k) * lj_dt) * lj_pres_old(q + lj_ny) + \]
\[ (-1.0\_sdk * lj_c2(k) * lj_r(k) * rt * qt * lj_dx) + \]
\[ lj_dy(k) * lj_dy(k) * lj_dt) * lj_ilj_old(q + lj_ny) \]

**ENDIF**
IF(i.eq.lj_nx) THEN
  lj_b(k+k+k-2) = lj_b(k+k+k-2) + ((lj_c0(k+lj_ny)+lj_c1(k+lj_ny))* & lj_r(k) * rb * gb * lj_dx * lj_dy(k) * lj_dy(k) * lj_dt) * & lj_pres_old(q-lj_ny) + (lj_c2(k+lj_ny) * lj_r(k) * rb * gb * & lj_dx * lj_dy(k) * lj_dy(k) * lj_dt) * lj_ilj_old(q-lj_ny)
ENDIF

ENDDO
ENDDO
END SUBROUTINE JetNonCon

END MODULE JetSetEqn

MODULE JetLUsolve
USE JetIntrType
!
!   This module contains some very old implementations of LINPACK
!   LU factorization/solution subroutines without reference to BLAS
!
CONTAINS

SUBROUTINE dgefa(a, lda, n, ipvt, info)
IMPLICIT NONE
!
!  Declaration Generated by genImpDecs.pl 5/98
INTEGER(sik) i, info, ip1, j, k, lda, m, n, nml
!
!  Declaration Generated by genImpDecs.pl 5/98
REAL(sdk) c, rp, t
!
!
!  because its name is same as routine in Cray system library.
!
!  dgefa factors a real matrix by gaussian elimination.
!
!  dgefa is usually called by dgeco, but it can be called
!  directly with a saving in time if rcond is not needed.
!  (time for dgeco) = (1 + 9/n) *(time for dgefa).
!
!  on entry
!    a    real(lda, n)
!         the matrix to be factored.
!    lda    integer
!         the leading dimension of the array a.
!    n    integer
!         the order of the matrix a.
!
!  on return
!    a    an upper triangular matrix and the multipliers

which were used to obtain it. 
the factorization can be written \( a = l \cdot u \) where
\( l \) is a product of permutation and unit lower 
triangular matrices and \( u \) is upper triangular.

ipvt integer(n)
 an integer vector of pivot indices. 

info integer
 = 0  normal value.
 = k if \( u(k,k) \cdot \text{eq.} \ 0.0 \). this is not an error 
condition for this subroutine, but it does 
indicate that dgesl or dgedit will divide by zero 
if called. use rcond in dgeco for a reliable 
indication of singularity.

this version is written in cdc compass for the ftn compiler 
communication convention. results may vary slightly from those 
obtained with a fortran version, and such a version is available 
upon request. this program simulates the fortran sequence..

subroutine dgefa (a, lda, n, ipvt, info)

INTEGER(sik), INTENT(OUT) :: ipvt(:)
INTEGER(sik) ipvt(n)

REAL(sdk), INTENT(INOUT) :: a(:,:)
REAL(sdk) a(lda,n)

info=0
IF (n.GT.1) THEN
 nml=n-1
 m=i
 ip1=i+1
 DO k=ip1,n
 m=k
 IF (abs(a(k,i)).GT.abs(a(m,i))) m=k
 ENDDO
 ipvt(i)=m
 IF (a(m,i).EQ.0.d0) THEN
 info=i
 ELSE
 IF (m.NE.i) THEN
 k=i,n
 t=a(i,k)
 a(i,k)=a(m,k)
 a(m,k)=t
 ENDDO
 ENDIF
 rp=-1.d0/a(i,i)
 DO k=ip1,n
 a(k,i)=a(k,i)*rp
 ENDDO
 DO j=ip1,n
 c=a(i,j)
 IF (c.NE.0.d0) THEN
 k=ip1,n
 a(k,j)=a(k,j)+a(k,i)*c
 ENDDO
 ENDIF
 ENDDO
```fortran
SUBROUTINE dgefa
IMPLICIT NONE
ipvt(n)=n
IF (a(n,n).EQ.0.d0) info=n
ELSE
  ipvt(1)=1
  IF (a(1,1).EQ.0.d0) info=1
ENDIF
RETURN
END SUBROUTINE dgefa

SUBROUTINE dgesl(a,lda,n,ipvt,b,job)
IMPLICIT NONE
!
!     Declaration Generated by genImpDecs.pl 5/98
INTEGER(sik) i,ipl,j,jml,job,k,kb,km1,kp1,l,lda,m,n,nml
!
!     Declaration Generated by genImpDecs.pl 5/98
REAL(sdk) c,t
!
because its name is same as routine in Cray system library.
!
dgesl solves the real system
a \cdot x = b \text{ or } \text{trans}(a) \cdot x = b
using the factors computed by dgeco or dgefa.
!
on entry
!
a      real(lda, n)
the output from dgeco or dgefa.
!
lda    integer
the leading dimension of the array a.
!
n      integer
the order of the matrix a.
!
ipvt   integer(n)
the pivot vector from dgeco or dgefa.
!
b      real(n)
the right hand side vector.
!
job    integer
= 0 to solve a\cdot x = b,
= nonzero to solve \text{trans}(a)\cdot x = b where
\text{trans}(a) is the transpose.
!
on return
!
b      the solution vector x.
!
error condition
!
a division by zero will occur if the input factor contains a
zero on the diagonal. technically this indicates singularity
but it is often caused by improper arguments or improper
setting of lda.
```
called correctly and if dgeco has set rcond .gt. 0.0
or dgefa has set info .eq. 0.

to compute \( \text{inverse}(a) \times c \) where \( c \) is a matrix
with \( p \) columns

```
call dgeco(a,lda,n,ipvt,rcond,z)
if (rcond is too small) go to ...
do 10 j = 1, p
   call dgesl(a,lda,n,ipvt,c(1,j),0)
10 continue
```

this version is written in cdc assembly language for the ftn
compiler communication convention. results may vary slightly
from those obtained with a fortran version, and such a version
is available upon request. this program simulates the
following fortran sequence..

```fortran
subroutine dgesl (a, lda, n, ipvt, b, job)
!
INTEGER(sik), INTENT(IN) :: ipvt(:)
INTEGER(sik) ipvt(n)
!
REAL(sdk), INTENT(INOUT) :: a(:,:),b(:)
REAL(sdk) a(lda,n),b(n)
!
nml=n-1
IF (n.LE.1) THEN
   b(1)=b(1)/a(1,1)
ELSEIF (job.EQ.0) THEN
   DO i=1,nm1
      m=ipvt(i)
      c=b(m)
      IF (m.NE.i) THEN
         b(m)=b(i)
         b(i)=c
      ENDIF
      IF (c.NE.0.d0) THEN
         ip1=i+1
         DO k=ip1,n
            b(k)=b(k)+a(k,i)*c
         ENDDO
      ENDIF
   ENDDO
ENDIF
ENDDO
DO i=1,nm1
   j=n-i+1
   b(j)=b(j)/a(j,j)
   c=b(j)
   IF (c.NE.0.d0) THEN
      jm1=j-1
      DO k=1,jm1
         b(k)=b(k)-a(k,j)*c
      ENDDO
   ENDIF
ENDDO
b(1)=b(1)/a(1,1)
ELSE
   DO k=1,n
      t=0.d0
      IF (k.NE.1) THEN
         km1=k-1
      ELSE
         km1=n-1
      ENDIF
      DO i=1,km1
         t=t+a(i,k)*b(i)
      ENDDO
      b(k)=t/a(k,k)
   ENDDO
```

! subroutine dgesl (a, lda, n, ipvt, b, job)
t = t + a(i,k) * b(i)
ENDDO
ENDDO
b(k) = (b(k) - t) / a(k,k)
ENDDO
DO kb = 1, n
k = n - kb
i = k + 1
DO i = k + 1, n
   t = t + a(i,k) * b(i)
ENDDO
b(k) = b(k) + t
l = IPVT(k)
IF (l .NE. k) THEN
   t = b(l)
   b(l) = b(k)
   b(k) = t
ENDIF
ENDDO
ENDIF
RETURN
END SUBROUTINE dgesl

REAL(sdk) FUNCTION ddot(n,dx,incx,dy,incy)
IMPLICIT NONE
!
!
! forms the dot product of two vectors.
!
! uses unrolled loops for increments equal to one.
!
! jack dongarra, linpack, 3/11/78.
!
REAL(sdk) dtemp
REAL(sdk), INTENT(INOUT) :: dx(:,), dy(:,)
INTEGER(sik) i, incx, incy, ix, iy, m, mp1, n
!
! ddot = 0.0d0
! dtemp = 0.0d0
IF (n .GT. 0) THEN
   IF (incx .EQ. 1 .AND. incy .EQ. 1) THEN
!
   ! code for both increments equal to 1
!
   ! clean-up loop
!
   n = MOD(n,5)
   IF (n .NE. 0) THEN
      DO i = 1, n
         dtemp = dtemp + dx(i) * dy(i)
      ENDDO
      IF (n .LT. 5) GOTO 60
   ENDIF
   mp1 = m + 1
   DO i = mp1, n
      dtemp = dtemp + dx(i) * dy(i) + dx(i+1) * dy(i+1) + dx(i+2) * dy(i+2) + dx(i+3) * dy(i+3) + dx(i+4) * dy(i+4)
   ENDDO
   60    ddot = dtemp
   ELSE
!
!
! code for unequal increments or equal increments
! not equal to 1
!
ix=1
iy=1
IF (incx.LT.0) ix=(-n+1)*incx+1
IF (incy.LT.0) iy=(-n+1)*incy+1
DO i=1,n
dtemp=dtemp+dx(ix)*dy(iy)
  ix=ix+incx
  iy=iy+incy
ENDDO
ddot=dtemp
ENDIF
ENDIF
RETURN
END FUNCTION ddot

SUBROUTINE dscal(n,da,dx,incx)
IMPLICIT NONE
!
!
! scales a vector by a constant.
! uses unrolled loops for increment equal to one.
! jack dongarra, linpack, 3/11/78.
!
REAL(sdk) da,dx(*)
INTEGER(sik) i,incx,m,mp1,n,nincx
!
IF (n.GT.0) THEN
  IF (incx.EQ.1) THEN
    !
    !        code for increment equal to 1
    !
    !        clean-up loop
    !
    m=mod(n,5)
    IF (m.NE.0) THEN
      DO i=1,m
        dx(i)=da*dx(i)
      ENDDO
    ELSE
      !
      !        code for increment not equal to 1
      !
      nincx=n*incx
      DO i=1,nincx,incx
        dx(i)=da*dx(i)
      ENDDO
      !
      !        code for increment equal to 1
      !
      !        clean-up loop
      !
      m=mod(n,5)
      IF (m.NE.0) THEN
        DO i=1,m
          dx(i)=da*dx(i)
        ENDDO
      ELSE
        !
        !        code for increment not equal to 1
        !
        nincx=n*incx
        DO i=1,nincx,incx
          dx(i)=da*dx(i)
        ENDDO
      !
      ELSE
        !
        !        code for increment equal to 1
        !
        !        clean-up loop
        !
        m=mod(n,5)
        IF (m.NE.0) THEN
          DO i=1,m
            dx(i)=da*dx(i)
          ENDDO
        ELSE
          !
          !        code for increment not equal to 1
          !
          nincx=n*incx
          DO i=1,nincx,incx
            dx(i)=da*dx(i)
          ENDDO
        !
        ELSE
          !
          !        code for increment equal to 1
          !
          !        clean-up loop
          !
          m=mod(n,5)
          IF (m.NE.0) THEN
            DO i=1,m
              dx(i)=da*dx(i)
            ENDDO
          ELSE
            !
            !        code for increment not equal to 1
            !
            nincx=n*incx
            DO i=1,nincx,incx
              dx(i)=da*dx(i)
            ENDDO
          !
          ELSE
            !
            !        code for increment equal to 1
            !
            !        clean-up loop
            !
            m=mod(n,5)
            IF (m.NE.0) THEN
              DO i=1,m
                dx(i)=da*dx(i)
              ENDDO
            ELSE
              !
              !        code for increment not equal to 1
              !
              nincx=n*incx
              DO i=1,nincx,incx
                dx(i)=da*dx(i)
              ENDDO
            !
            ELSE
              !
              !        code for increment equal to 1
              !
              !        clean-up loop
              !
              m=mod(n,5)
              IF (m.NE.0) THEN
                DO i=1,m
                  dx(i)=da*dx(i)
                ENDDO
              ELSE
                !
                !        code for increment not equal to 1
                !
                nincx=n*incx
                DO i=1,nincx,incx
                  dx(i)=da*dx(i)
                ENDDO
              !
            ENDIF
          ENDDO
        !
      ENDIF
    ENDDO
  ENDIF
ENDIF
RETURN
END SUBROUTINE dscal
SUBROUTINE daxpy(n, da, dx, incx, dy, incy)
IMPLICIT NONE

! constant times a vector plus a vector.
! uses unrolled loops for increments equal to one.
! jack dongarra, linpack, 3/11/78.
!
REAL(sdk), INTENT(INOUT) :: dx(:), dy(:), da
REAL(sdk) dx(*), dy(*), da
INTEGER(sik) i, incx, incy, ix, iy, m, mp1, n
!
IF (n.GT.0) THEN
  IF (da.NE.0.0d0) THEN
    IF (incx.EQ.1.AND.incy.EQ.1) THEN
      ! code for both increments equal to 1
      !
      ! clean-up loop
      !
      m=mod(n,4)
      IF (m.NE.0) THEN
        DO i=1,m
          dy(i)=dy(i)+da*dx(i)
        ENDDO
      ENDIF
      mp1=m+1
      DO i=mp1,n,4
        dy(i)=dy(i)+da*dx(i)
        dy(i+1)=dy(i+1)+da*dx(i+1)
        dy(i+2)=dy(i+2)+da*dx(i+2)
        dy(i+3)=dy(i+3)+da*dx(i+3)
      ENDDO
    ELSE
      ! code for unequal increments or equal increments
      ! not equal to 1
      !
      ix=1
      iy=1
      IF (incx.LT.0) ix=-(n+1)*incx+1
      IF (incy.LT.0) iy=-(n+1)*incy+1
      DO i=1,n
        dy(iy)=dy(iy)+da*dx(ix)
        ix=ix+incx
        iy=iy+incy
      ENDDO
    ENDIF
  ELSE
    ! code for unequal increments or equal increments
    ! not equal to 1
    !
    ix=1
    iy=1
    IF (incx.LT.0) ix=-(n+1)*incx+1
    IF (incy.LT.0) iy=-(n+1)*incy+1
    DO i=1,n
      dy(iy)=dy(iy)+da*dx(ix)
      ix=ix+incx
      iy=iy+incy
    ENDDO
  ENDIF
ENDIF
ENDIF
ENDIF
51 RETURN
END SUBROUTINE daxpy
INTEGER(sik) FUNCTION idamax(n,dx,incx)
IMPLICIT NONE
!
! finds the index of element having max. absolute value.
! jack dongarra, linpack, 3/11/78.
!
REAL(sdk) dx(*),dmax
INTEGER(sik) i,incx,ix,n
!
idamax=0
IF (n.GE.1) THEN
  idamax=1
IF (n.NE.1) THEN
  IF (incx.EQ.1) THEN
    ! code for increment equal to 1
    dmax=dabs(dx(1))
    DO i=2,n
      IF (dabs(dx(i)).GT.dmax) THEN
        idamax=i
        dmax=dabs(dx(i))
      ENDIF
    ENDDO
  ELSE
    ! code for increment not equal to 1
    ix=1
    dmax=dabs(dx(1))
    ix=ix+incx
    DO i=2,n
      IF (dabs(dx(ix)).GT.dmax) THEN
        idamax=i
        dmax=dabs(dx(ix))
      ENDIF
      ix=ix+incx
    ENDDO
  ENDIF
ENDIF
ENDIF
RETURN
END FUNCTION idamax

SUBROUTINE dgbfa(abd,lda,n,ml,mu,ipvt,info)
IMPLICIT NONE
!
!***begin prologue  dgbfa
!***date written   780814   (yymmdd)
!***revision date  861211   (yymmdd)
!***category no.  d2a2
!***keywords  library=slatec(linpack),
!             type=double precision(sgbfa-s dgbfa-d cgbfa-c),banded,
!             linear algebra,matrix,matrix factorization
!***author  moler, c. b., (u. of new mexico)
!***purpose  factors a double precision band matrix by elimination.
!***description
!
  dgbfa factors a double precision band matrix by elimination.
dgbfa is usually called by dgbc0, but it can be called
directly with a saving in time if rcond is not needed.

on entry

abdi double precision(lda, n)
contains the matrix in band storage. the columns
of the matrix are stored in the columns of abdi and
the diagonals of the matrix are stored in rows
ml+1 through 2*ml+mu+1 of abdi.
see the comments below for details.

lda integer
the leading dimension of the array abdi.
lda must be .ge. 2*ml + mu + 1.

n integer
the order of the original matrix.

ml integer
number of diagonals below the main diagonal.
0 .le. ml .lt. n.

mu integer
number of diagonals above the main diagonal.
0 .le. mu .lt. n.
more efficient if ml .le. mu.

on return

abdi an upper triangular matrix in band storage and
the multipliers which were used to obtain it.
the factorization can be written a = l*u where
l is a product of permutation and unit lower
triangular matrices and u is upper triangular.

ipvt integer(n)
an integer vector of pivot indices.

info integer
= 0 normal value.
= k if u(k,k) .eq. 0.0. this is not an error
condition for this subroutine, but it does
indicate that dgbs1 will divide by zero if
called. use rcond in dgbc0 for a reliable
indication of singularity.

band storage

if a is a band matrix, the following program segment
will set up the input.

ml = (band width below the diagonal)
mu = (band width above the diagonal)
m = ml + mu + 1

do 20 j = 1, n
   il = max0(1, j-mu)
i2 = min0(n, j+ml)
   do 10 i = il, i2
      k = i - j + m
      abdi(k,j) = a(i,j)

   10 continue

20 continue
this uses rows ml+1 through 2*ml+mu+1 of abd.
in addition, the first ml rows in abd are used for
elements generated during the triangularization.
the total number of rows needed in abd is 2*ml+mu+1.
the ml+mu by ml+mu upper left triangle and the
ml by ml lower right triangle are not referenced.

linpack. this version dated 08/14/78.
cleve moler, university of new mexico, argonne national lab.

subroutines and functions
blas daxpy,dscal,idamax
fortran max0,min0
***references dongarra j.j., bunch j.r., moler c.b., stewart g.w.,
***routines called daxpy,dscal,idamax
***end prologue dgbfa
INTEGER(sik) lda,n,ml,mu,ipvt(*),info
REAL(sdk) abd(lda,n)
!
REAL(sdk) t
INTEGER(sik) i,i0,j,ju,jz,j0,j1,k,kp1,l,lm,m,mm,nm1

***first executable statement dgbfa
m=ml+mu+1
info=0

! zero initial fill-in columns
!
  j0=mu+2
  j1= min(n,m)-1
  IF (j1.GE.j0) THEN
    DO jz=j0,j1
      i0=m+1-jz
      DO i=i0,ml
        abd(i,jz)=0.0d0
      ENDDO
    ENDDO
  !
  jz=j1
  ju=0
 !
  gaussian elimination with partial pivoting
  !
  nm1=n-1
  IF (nm1.GE.1) THEN
    DO k=1,nm1
      kp1=k+1
    !
    zero next fill-in column
    !
      jz=jz+1
      IF (jz.LE.n) THEN
        IF (ml.GE.1) THEN
          DO i=1,ml
            abd(i,jz)=0.0d0
          ENDDO
ENDDIF
ENDIF
!
find l = pivot index
!
  lm= min(ml,n-k)
l=idamax(lm+1,abd(m,k),1)+m-1
ipvt(k)=l+k-m
!
zero pivot implies this column already triangularized
!
IF (abd(l,k).EQ.0.0d0) THEN
  info=k
ELSE
  interchange if necessary
  !
  IF (l.NE.m) THEN
    t=abd(l,k)
    abd(l,k)=abd(m,k)
    abd(m,k)=t
  ENDIF
  !
  compute multipliers
  !
  t=-1.0d0/abd(m,k)
  CALL dscal(lm,t,abd(m+1,k),1)
  !
  row elimination with column indexing
  !
  ju= min( max(ju,mu+ipvt(k)),n)
  mm=m
  IF (ju.GE.kp1) THEN
    DO j=kp1,ju
       l=l-1
       mm=mm-1
       t=abd(l,j)
       IF (l.NE.mm) THEN
          abd(l,j)=abd(mm,j)
          abd(mm,j)=t
       ENDIF
       CALL daxpy(lm,t,abd(m+1:m+lm,k),1,abd(mm+1:mm+lm,j),1)
    ENDDO
  ENDIF
ENDIF
ENDDO
ENDIF
ENDDO
ENDIF
ipvt(n)=n
IF (abd(m,n).EQ.0.0d0) info=n
RETURN
END SUBROUTINE dgbfa

SUBROUTINE dgbsl(abd,lda,n,ml,mu,ipvt,b,job)
IMPLICIT NONE
!
***begin prologue  dgbsl
***date written   780814   (yymmdd)
***revision date  861211   (yymmdd)
***category no.  d2a2
***keywords  library=slatec(linpack),
            type=double precision(sgbsl-s dgbsl-d cgbsl-c),banded,
! linear algebra, matrix, solve
!***author moler, c. b., (u. of new mexico)
!***purpose solves the double precision band system a*x=b or
! trans(a)*x=b using the factors computed by dgbco or dgbfa.
!***description
!
dgbsl solves the double precision band system
a * x = b or trans(a) * x = b
using the factors computed by dgbco or dgbfa.
!
on entry
!
   abd double precision(lda, n)
   the output from dgbco or dgbfa.
!
   lda integer
   the leading dimension of the array abd.
!
   n integer
   the order of the original matrix.
!
   ml integer
   number of diagonals below the main diagonal.
!
   mu integer
   number of diagonals above the main diagonal.
!
   ipvt integer(n)
   the pivot vector from dgbco or dgbfa.
!
   b double precision(n)
   the right hand side vector.
!
   job integer
   = 0 to solve a*x = b,
   = nonzero to solve trans(a)*x = b, where
       trans(a) is the transpose.
!
on return
!
   b the solution vector x.
!
error condition
!
   a division by zero will occur if the input factor contains a
   zero on the diagonal. technically this indicates singularity
   but it is often caused by improper arguments or improper
   setting of lda. it will not occur if the subroutines are
   called correctly and if dgbco has set rcond .gt. 0.0
   or dgbfa has set info .eq. 0.
!
   to compute inverse(a) * c where c is a matrix
   with p columns
   call dgbco(abd,lda,n,ml,mu,ipvt,rcond,z)
   if (rcond is too small) go to ...
   do 10 j = 1, p
      call dgbsl(abd,lda,n,ml,mu,ipvt,c(1,j),0)
   10 continue
!
linpack. this version dated 08/14/78.
cleve moler, university of new mexico, argonne national lab.
subroutines and functions
blas daxpy, ddot
fortran min0

***references dongarra j.j., bunch j.r., moler c.b., stewart g.w.,
***routines called daxpy, ddot
***end prologue dgbsl

INTEGER(sik) lda,n,ml,mu,job
INTEGER(sik), DIMENSION(:) :: ipvt
REAL(sdk), DIMENSION(:) :: b
REAL(sdk), DIMENSION(:,:) :: abd
REAL(sdk) t
INTEGER(sik) k,kb,l,la,lb,lm,m,nm1

**first executable statement dgbsl
m=mu+ml+1
nml=n-1
IF (job.EQ.0) THEN
!
!        job = 0 , solve a * x = b
!        first solve l*y = b
!
IF (ml.NE.0) THEN
IF (nm1.GE.1) THEN
DO k=1,nm1
lm= min(ml,n-k)
l=ipvt(k)
t=b(l)
IF (l.NE.k) THEN
b(l)=b(k)
b(k)=t
ENDIF
CALL daxpy(lm,t,abd(m+1:,k),1,b(k+1:),1)
ENDDO
ENDIF
!
!        now solve u*x = y
!
DO kb=1,n
k=n+1-kb
b(k)=b(k)/abd(m,k)
lm= min(k,m)-1
la=m-lm
lb=k-lm
t=-b(k)
CALL daxpy(lm,t,abd(la:,k),1,b(lb:),1)
ENDDO
ELSE
!
!        job = nonzero, solve trans(a) * x = b
!        first solve trans(u)*y = b
!
DO k=1,n
lm= min(k,m)-1
la=m-lm
lb=k-lm
t=ddot(lm,abd(la:,k),1,b(lb:),1)
b(k)=(b(k)-t)/abd(m,k)
ENDDO
SUBROUTINE dgbsl

END SUBROUTINE dgbsl

END MODULE JetLUsolve

MODULE JetSetVol

USE JetIntrType
USE JetDataIn

CONTAINS

SUBROUTINE JetCellSize

END SUBROUTINE JetCellSize

USE JetIntrType
USE JetDataIn

CONTAINS

SUBROUTINE JetCellSize
INTEGER :: i, j, k, l
REAL :: r

! Set the size of the computational volume in the radial direction
DO i=1,lj_nx
  DO j=1,lj_nylj
    k=j+(i-1)*lj_ny
    ! For the liquid jet
    lj_dy(k)=lj_rlj(i)/lj_nylj
  ENDDO
  DO j=lj_nylj+1,lj_ny
    k=j+(i-1)*lj_ny
    ! For the vapor boundary layer
    lj_dy(k)=(lj_rvb-lj_rlj(i))/lj_nyvb
  ENDDO
ENDDO

! Set the size of computational volume for radial momentum equation
DO i=1,lj_nx
  DO j=1,lj_nylj
    k=j+(i-1)*(lj_ny+1)
    l=j+(i-1)*lj_ny
    lj_dyy(k)=lj_dy(l)
  ENDDO
  j=lj_nylj+1
  k=j+(i-1)*(lj_ny+1)
  lj_dyy(k)=lj_dyy(k-1)
  DO j=lj_nylj+2,lj_ny
    k=j+(i-1)*(lj_ny+1)
    l=j+(i-1)*lj_ny
    lj_dyy(k)=lj_dy(l)
  ENDDO
  j=lj_ny+1
  k=j+(i-1)*(lj_ny+1)
  lj_dyy(k)=lj_dyy(k-1)
ENDDO

DO i=1,lj_nx
  DO j=1,lj_ny
    k=j+(i-1)*lj_ny
    lj_dyx(k)=lj_dy(k)
  ENDDO
ENDDO

i=lj_nx+1
DO j=1,lj_ny
  k=j+(i-1)*lj_ny
  lj_dyx(k)=lj_dyx(k-lj_ny)
ENDDO

! Set the jet radius size for continuity cells
DO i=1,lj_nx
  DO j=1,lj_ny
    k=j+(i-1)*lj_ny
    ! Within the liquid jet
    lj_r(k)=(j-0.5_sdk)*lj_dy(k)
  ENDDO
  DO j=lj_ny+1,lj_ny
    k=j+(i-1)*lj_ny
    ! Within the vapor boundary layer
    lj_r(k)=lj_rlj(i)+(j-lj_ny-0.5_sdk)*lj_dy(k)
  ENDDO
ENDDO

! Set the jet radius for axial momentum cells
DO i=2,lj_nx
  DO j=1,lj_ny
    k=j+(i-1)*lj_ny
    lj_rx(k)=0.5_sdk*(lj_r(k)+lj_r(k-lj_ny))
  ENDDO
ENDDO

i=lj_nx+1
DO j=1,lj_ny
  k=j+(i-1)*lj_ny
  lj_rx(k)=lj_r(k-lj_ny)
ENDDO

i=1
DO j=1,lj_ny
  k=j+(i-1)*lj_ny
  lj_rx(k)=lj_r(k)
ENDDO

! Set the jet radius for radial momentum cells
DO i=1,lj_nx
  DO j=1,lj_ny+1
    k=j+(i-1)*(lj_ny+1)
    lj_ry(k)=(j-1.0)*lj_dyy(k)
  ENDDO
  DO j=lj_ny+2,lj_ny+1
    k=j+(i-1)*(lj_ny+1)
    lj_ry(k)=lj_rlj(i)+(j-(lj_ny+1))*lj_dyy(k)
  ENDDO
ENDDO

! Set the computational cell volume size
DO i=1,lj_nx
  DO j=1
    k=j+(i-1)*lj_ny
    lj_vol(k)=lj_pi*lj_dy(k)*lj_dy(k)*lj_dx
  ENDDO
DO  j=2,lj_nylj
    k=j+(i-1)*lj_ny
    lj_vol(k)=lj_pi*(j+j-1)*lj_dy(k)*lj_dy(k)*lj_dx
ENDDO

DO  j=lj_nylj+1,lj_ny
    k=j+(i-1)*lj_ny
    l=j-lj_nylj
    lj_vol(k)=lj_pi*((l+l-1)*lj_dy(k)*lj_dy(k)+lj_rlj(i)*lj_dy(k)+lj_rlj(i)*lj_dy(k))*lj_dx
ENDDO

!  Set x-momentum computational cell volume size
DO  i=1,lj_nx
  DO  j=1,lj_ny
    k=j+(i-1)*lj_ny
    lj_volx(k)=lj_vol(k)
  ENDDO
ENDDO

DO  j=1,lj_ny
  i=lj_nx+1
  k=j+(i-1)*lj_ny
  lj_volx(k)=lj_vol(k-lj_ny)
ENDDO

!  Set y-momentum computational cell volume size
DO  i=1,lj_nx
  j=1
  k=j+(i-1)*(lj_ny+1)
  lj_voly(k)=0.25*lj_pi*lj_dyy(k)*lj_dyy(k)*lj_dx
  DO  j=2,lj_nylj+1
    k=j+(i-1)*(lj_ny+1)
    lj_voly(k)=lj_pi*(j+j-2)*lj_dyy(k)*lj_dyy(k)*lj_dx
  ENDDO
  DO  j=lj_nylj+2,lj_ny+1
    k=j+(i-1)*(lj_ny+1)
    l=j-lj_nylj-1
    lj_voly(k)=lj_pi*((l+l-2)*lj_dyy(k)*lj_dyy(k)+lj_rlj(i)*lj_dyy(k)+lj_rlj(i)*lj_dyy(k))*lj_dx
  ENDDO
ENDDO

!  Set Cell Face areas for mass balance
!  Set Axial faces of continuity cells

DO  i=1,lj_nx+1
  DO  j=1,lj_nylj
    k=j+(i-1)*lj_ny
    lj_axlj(k)=lj_pi*lj_dyx(k)*lj_dyx(k)*(j+j-1)
  ENDDO
ENDDO

DO  j=1,lj_nylj+1,lj_ny
  k=j+(i-1)*lj_ny
  lj_axlj(k)=lj_pi*lj_dyx(k)*lj_dyx(k)*(j+j-1)
ENDDO
248

l=j-lj_nylj
IF(i.eq.lj_nx+1)THEN
  r=lj_rlj(i-1)
ELSE
  r=lj_rlj(i)
ENDIF

lj_axlj(k)=lj_pi*(2.0_sdk*r*lj_dyx(k)+lj_dyx(k)*lj_dyx(k)*(l+l-1))
ENDDO
ENDDO

! Set radial faces of continuity cells
DO i=1,lj_nx
  DO j=1,lj_nylj+1
    k=j+(i-1)*lj_ny+1
    lj_aylj(k)=2.0_sdk*lj_pi*(j-1)*lj_dyy(k)*lj_dx
  ENDDO
  DO j=lj_nylj+2,lj_ny+1
    k=j+(i-1)*lj_ny+1
    l=j-(lj_nylj+1)
    lj_aylj(k)=2.0_sdk*lj_pi*(lj_rlj(i)+l*lj_dyy(k))*lj_dx
  ENDDO
ENDDO
ENDDO

END SUBROUTINE JetCellSize

SUBROUTINE JetProps
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! This subroutine calculates the properties of the jet and vapor boundary layer before each time-step. !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
IMPLICIT NONE

INTEGER :: instat,icrit,iregn,iflag,i,j,k,q
REAL :: drdp,drdh,dtdp,dtdh,dvdp,dvdh,lj_props(10)

! lj_ilj=18.1086928779532_sdk
! lj_rholj=62.4096422835778_sdk
! lj_cp=1.00206440157967_sdk

CALL WPINIT(0,0,0,12,instat)

DO i=1,lj_nx
  DO j=1,lj_nylj
    k=j+(i-1)*lj_ny+1
    q=(j+1)+i*(lj_ny+2)
    CALL RHOTLQ(lj_pres(k),lj_ilj(k),lj_props,lj_rholj(q),lj_tlj(k),drdp,drdh,dtdp,dtdh,dvdp,dvdh,lj_st(k),lj_kf(k),lj_visc(k),iregn,icrit,iflag)
    lj_cp(k)=1.0_sdk/dtdh
    lj_drhodi(k)=drdh
    lj_drhodp(k)=drdp
    lj_if(k)=lj_props(3)
  ENDDO
ENDDO

END SUBROUTINE JetProps
lj_ig(k) = lj_props(4)
lj_tsat(k) = lj_props(5)
lj_difdp(k) = lj_props(8)
ENDDO

DO i = 1, lj_nx
  DO j = lj_nylj + 1, lj_ny
    k = j + (i - 1) * lj_ny
    q = (j + 1) + i * (lj_ny + 2)
    CALL RHOTVP(lj_pres(k), lj_ilj(k), lj_props, lj_rholj(q), lj_tlj(k), &
               lj_drhodp(k), &
               lj_if(k) = lj_props(3)
               lj_ig(k) = lj_props(4)
               lj_tsat(k) = lj_props(5)
               lj_difdp(k) = lj_props(8)
ENDDO

lj_cp(k) = 1.0_sdk / dtdh
lj_drhodi(k) = drdh
lj_drhodp(k) = drdp
lj_if(k) = lj_props(3)
lj_ig(k) = lj_props(4)
lj_tsat(k) = lj_props(5)
lj_difdp(k) = lj_props(8)
ENDDO

! Set Dummy value at j=0
DO i = 1, lj_nx
  q = 1 + i * (lj_ny + 2)
  k = 2 + i * (lj_ny + 2)
  lj_rholj(q) = lj_rholj(k)
ENDDO

! Set Dummy value at j=lj_ny+1
DO i = 1, lj_nx
  q = lj_ny + 2 + i * (lj_ny + 2)
  k = lj_ny + 1 + i * (lj_ny + 2)
  j = lj_ny + i * (lj_ny + 2)
  lj_rholj(q) = 2.0_sdk * lj_rholj(k) - lj_rholj(j)
ENDDO

! Set Dummy value at i=0
DO j = 1, lj_ny
  q = (j + 1) + 0 * (lj_ny + 2)
  k = (j + 1) + 1 * (lj_ny + 2)
  i = (j + 1) + 2 * (lj_ny + 2)
  lj_rholj(q) = 2.0_sdk * lj_rholj(k) - lj_rholj(i)
ENDDO

! Set Dummy value at i=lj_nx+1
DO j = 1, lj_ny
  q = (j + 1) + (lj_nx + 1) * (lj_ny + 2)
  k = (j + 1) + (lj_nx) * (lj_ny + 2)
  i = (j + 1) + (lj_nx - 1) * (lj_ny + 2)
  lj_rholj(q) = 2.0_sdk * lj_rholj(k) - lj_rholj(i)
ENDDO

END SUBROUTINE JetProps
SUBROUTINE JetOneSet

IMPLICIT NONE

REAL, ALLOCATABLE :: lj_vol_av(:), lj_i_av(:), lj_u_av(:), lj_t_av(:)
INTEGER :: i, j, k, l

ALLOCATE(lj_vol_av(lj_nx), lj_i_av(lj_nx), lj_u_av(lj_nx), lj_t_av(lj_nx))

lj_vol_av=0.0_sdk
lj_i_av=0.0_sdk
lj_u_av=0.0_sdk
lj_t_av=0.0_sdk
lj_tv=0.0_sdk
lj_volv=0.0_sdk

DO i=1, lj_nx
  DO j=1, lj_ny
    k=(j+1) * lj_sm
    lj_i_av(i)=lj_i_av(i)+lj_pi*(j+j-1)*lj_dy(k)*lj_dy(k)*lj_ilj(k)
    lj_t_av(i)=lj_t_av(i)+lj_pi*(j+j-1)*lj_dy(k)*lj_dy(k)*lj_tlj(k)
    lj_vol_av(i)=lj_vol_av(i)+lj_pi*(j+j-1)*lj_dy(k)*lj_dy(k)
    lj_u_av(i)=lj_u_av(i)+lj_pi*(j+j-1)*lj_dy(k)*lj_dy(k)*lj_ulj(k)
    lj_tv(i)=lj_tv(i)+lj_pi*((j+j-1)*lj_dy(k)*lj_dy(k)+lj_rlj(i)*lj_dy(k))*lj_tlj(k)
    lj_volv(i)=lj_volv(i)+lj_pi*((j+j-1)*lj_dy(k)*lj_dy(k)+lj_rlj(i)*lj_dy(k))
  ENDDO
ENDDO

lj_i_av=lj_i_av/lj_vol_av
lj_t_av=lj_t_av/lj_vol_av
lj_u_av=lj_u_av/lj_vol_av
lj_tv=lj_tv/lj_volv
lj_thumb=lj_i_av
lj_t=lj_t_av
lj_u=lj_u_av

DO i=1, lj_nx
  WRITE(15,'(6F10.4)') lj_i_av(i), lj_t_av(i), lj_u_av(i), lj_tlj(k), lj_ilj(k), lj_ulj(k)
ENDDO

END SUBROUTINE JetOneSet

SUBROUTINE JetDiffusivity

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Subroutine Jet Diffusivity will calculate the thermal diffusivity
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
END SUBROUTINE JetDiffusivity
! diffusivity and the momentum diffusivity. The diffusivities can be either turbulent or found using the properties.

IMPLICIT NONE

INTEGER(sik) :: i,j,k,q,l
REAL(sdk) :: nu(lj_n),edm(lj_n),alf(lj_n),dvg(lj_n),edh(lj_n),edg(lj_n)

nu  = 0.0_sdk
edm = 0.0_sdk
alf = 0.0_sdk
edh = 0.0_sdk
dvg = 0.0_sdk
edg = 0.0_sdk

DO i=1,lj_nx
    DO j=1,lj_ny
        k=j+(i-1)*lj_ny
        q=(j+1)+i*(lj_ny+2)
        nu(k)=lj_visc(k)/lj_rholj(q)
        alf(k)=lj_kf(k)/(lj_rholj(q)*lj_cp(k))
    ENDDO
ENDDO

vg = 0.00002_sdk  ! coefficient of diffusion m^2/s

DO i=1,lj_nx
    DO j=1,lj_nylj
        k=j+(i-1)*(lj_ny+1)
        edm(k)=0.0013*lj_uin*lj_din
        edh(k)=1.3*edm(k)
        edg(k)=edm(k)
    ENDDO
ENDDO

lj_edm=0.0_sdk
lj_edh=0.0_sdk
lj_edh=edh+alf
lj_edm=edm+nu
lj_edg=edg+dvg

! Set eddy diffusivity for y-momentum equation
DO i=1,lj_nx
    DO j=1,lj_nylj
        k=j+(i-1)*(lj_ny+1)
l=j+(i-1)*lj

lj_edmy(k)=lj_edm(l)
ENDDO

DO j=lj_nylj+2,lj
k=j+(i-1)*(lj+1)
l=j+(i-1)*lj

lj_edmy(k)=lj_edm(l)
ENDDO

j=lj_nylj+1
k=j+(i-1)*(lj+1)
l=lj_nylj+(i-1)*lj

lj_edmy(k)=lj_edm(l)

j=lj_ny+1
k=j+(i-1)*(lj+1)
lj_edmy(k)=lj_edm(k-1)
ENDDO

DO i=2,lj
DO j=1,lj

lj_edmx(k)=0.5*(lj_edm(k)+lj_edm(k-lj))

ENDDO
ENDDO

DO i=2,lj
DO j=1,lj

lj_edmx(k)=lj_edm(k)
ENDDO

END SUBROUTINE JetDiffusivity

END SUBROUTINE Echo
!
! This subroutine echos the input to an output file
INTEGER(sik) :: i

CHARACTER(45) :: cu0, cd0, ct0, cts, cl, cpres, cnylj, cnyvb, cnx
CHARACTER(45) :: ctstart, ctend, cdvb, cstep

OPEN(36, FILE='jet.out')
OPEN(16, FILE='jetdetail.out')

cu0 = 'The initial jet velocity is (ft/s)'
ct0 = 'The initial jet temperature is (F)'
cl  = 'The jet length is (ft)'
cts = 'The steam temperature is F)'
cl  = 'The jet length is (ft)'
ct0 = 'The initial jet temperature is (F)'
cl  = 'The jet length is (ft)'
cts = 'The steam temperature is F)'
cl  = 'The jet length is (ft)'
cts = 'The steam temperature is F)'
ct0 = 'The initial jet temperature is (F)'
ct0 = 'The initial jet temperature is (F)'
cl  = 'The jet length is (ft)'
cts = 'The steam temperature is F)'

ctstart = 'The start time is'
ctend = 'The end time is'
cstep = 'The maximum time step size is'

WRITE(36,22) cu0, ceq, lj_uin
WRITE(36,22) cd0, ceq, lj_din
WRITE(36,22) ct0, ceq, lj_tin
WRITE(36,22) cts, ceq, lj_tstm
WRITE(36,22) cl, ceq, lj_length
WRITE(36,22) cpres, ceq, lj_pin
WRITE(36,44) cnylj, ceq, lj_nylj
WRITE(36,44) cnyvb, ceq, lj_nyvb
WRITE(36,44) cnx, ceq, lj_nx
WRITE(36,22) cdvb, ceq, lj_dvb
WRITE(36,22) cstep, ceq, lj_tmax

! END SUBROUTINE Echo

SUBROUTINE Edit

! This subroutine prints out the liquid jet model data in a graphics friendly manner
! Output name is temps.out.

IMPLICIT NONE

INTEGER(sik) :: i, j, k, q
REAL(sdk) :: xc

OPEN(11, FILE='jetout.csv', POSITION='APPEND')
OPEN(22, FILE='jetout.dat')
! OPEN(25,FILE='jetout2.csv',POSITION='APPEND')
!
! WRITE(11,'(F6.4,a,I2,a,F6.4)') test_num, '   ',n, '   ',ttime
!
! FORMAT(F10.5,lj_lEN20.6,lj_mEN20.6,lj_nF15.6,lj_nF15.6)
!
! WRITE(11,*) 'Time is ', ttime
!
! WRITE(11,*) ttime, (lj_ulj(k),k=1,lj_l),(lj_vlj(k),k=1,lj_m),
! (lj_tlj(k),k=1,lj_n),(lj_ilj(k),k=1,lj_n)
WRITE(11,'(F10.5,a)',ADVANCE='no') ttime, ' ',
WRITE(22,'(F10.5,a)',ADVANCE='no') 0.0*lj_dx, lj_tin
DO i=1,lj_l
   WRITE(11,'(F30.10,a)',ADVANCE='no') lj_ulj(i), ' ',
ENDDO
DO j=1,lj_m
   WRITE(11,'(F30.10,a)',ADVANCE='no') lj_vlj(j), ' ',
ENDDO
DO k=1,lj_n
   WRITE(11,'(F30.10,a)',ADVANCE='no') lj_tlj(k), ' ',
ENDDO
DO k=1,lj_n
   WRITE(11,'(F30.10,a)',ADVANCE='no') lj_ilj(k), ' ',
ENDDO
DO k=1,lj_n
   WRITE(11,'(F30.10,a)',ADVANCE='no') lj_pres(k), ' ',
ENDDO
DO i=1,lj_nx
   WRITE(11,'(F30.10,a)',ADVANCE='no') lj_t(i), ' ',
      WRITE(22,'(F10.5,4x,F30.10)') (i-0.5)*lj_dx, lj_t(i)
ENDDO
DO i=1,lj_nx
   j=lj_nylj
   q=j+i*(lj_ny+2)
   WRITE(11,'(F30.10,a)',ADVANCE='no') lj_glj(q), ' ',
ENDDO
DO i=1,lj_nx
   k=lj_nylj+(i-1)*lj_ny
   WRITE(11,'(F30.10,a)',ADVANCE='no') lj_qil(k), ' ',
ENDDO
!
DO i=1,lj_nx
   WRITE(25,'(F30.10,a)',ADVANCE='no') lj_hil(i), ' ',
! ENDDO
CLOSE(22)
CLOSE(11)
END SUBROUTINE Edit
SUBROUTINE PrintOut

IMPLICIT NONE

INTEGER(sik) :: i, j, k, q

WRITE(36, *) "Time", ttime
WRITE(36, ' (a)') 'Axial Position  Jet Radius  Jet Velocity  Jet Temperature
Vapor Temperature'
WRITE(36, ' (a)') ' x(ft)     R(ft)     u(ft/s)     T(F)     T(F)''
WRITE(36, ' (F10.5,4x,F10.5,3x,F10.5,7x,F10.5,F10.5)') 0.0*lj_dx, lj_rlj(1), lj_u(1), lj_tin, lj_tstm

DO i=1,lj_nx
   WRITE(36, ' (F10.5,4x,F10.5,3x,F10.5,7x,F10.5,7x,F10.5)') (i-0.5)*lj_dx, lj_rlj(i), lj_u(i), lj_t(i), lj_tv(i)
ENDDO
WRITE(36, ' (/)')

WRITE(16, *) "Time", ttime
WRITE(16, ' (a)') 'I       J            velocity''
DO i=1,lj_nx+1
   DO j=1,lj_ny
      k=j+(i-1)*lj_ny
      WRITE(16, ' (I4,4x,I4,4x,F20.10)')  i,j,lj_ulj(k)
   ENDDO
ENDDO
WRITE(16, ' (a)') 'I       J         velocity''
DO i=1,lj_nx
   DO j=1,lj_ny+1
      k=j+(i-1)*(lj_ny+1)
      WRITE(16, ' (I4,4x,I4,4x,F12.5)') i,j,lj_vlj(k)
   ENDDO
ENDDO
WRITE(16, ' (a)') 'I       J        enthalpy''
DO i=1,lj_nx
   DO j=1,lj_ny
      k=j+(i-1)*lj_ny
      WRITE(16, ' (I4,4x,I4,4x,F12.5)') i,j,lj_ilj(k)
   ENDDO
ENDDO
WRITE(16, ' (a)') 'I       J        pressure''
DO i=1,lj_nx
   DO j=1,lj_ny
      k=j+(i-1)*lj_ny
      WRITE(16, ' (I4,4x,I4,4x,F12.5)') i,j,lj_ilj(k)
   ENDDO
ENDDO
WRITE(16, ' (a)') 'I       J        pressure''
DO i=1,lj_nx
   DO j=1,lj_ny
      k=j+(i-1)*lj_ny
      WRITE(16, ' (I4,4x,I4,4x,F12.5)') i,j,lj_ilj(k)
   ENDDO
ENDDO
WRITE(16,'(I4,4x,I4,4x,F12.5)') i,j,lj_pres(k)
ENDDO

WRITE(16,'(a)') '   I       J        density'
DO i=1,lj_nx
  DO j=1,lj_ny
    q=j+i*(lj_ny+2)
    WRITE(16,'(I4,4x,I4,4x,F12.5)') i,j,lj_rholj(q)
  ENDDO
ENDDO
WRITE(16,'(/)')
END SUBROUTINE PrintOut

END MODULE JetOutput

MODULE JetHeatTrans

!*****************************************************************************!
!                                                                             !
!     This module contains the heat and mass transfer models for use in the    !
!       liquid jet condensation model. Heat transfer models have been         !
!       moved to separate module from SetEqns.                               !
!                                                                             !
!     Programmed by Frank Buschman                                            !
!        rev. 0 4/2007                                                        !
!        rev. 1 5/15/2007                                                     !
!*****************************************************************************!

USE JetIntrType
USE JetDatain

CONTAINS

SUBROUTINE liqsideh

!*****************************************************************************!
!                                                                             !
!     Subroutine to set liquid side heat transfer coefficient                  !
!                                                                             !
!     Programmed by Frank Buschman                                            !
!        rev. 0 5/15/2007                                                     !
!*****************************************************************************!

IMPLICIT NONE

INTEGER(sik) :: i,j,k,l
REAL(sdk) :: tvb_av(lj_nx),vb_area(lj_nx),lj_tsurf(lj_nx)
REAL(sdk) :: lj_hil_lam(lj_nx),lj_hil_turb(lj_nx),rho,visc,ulam
REAL(sdk) :: lj_renew_time, lj_afact, lj_cfact
!INTEGER :: iphase,iregn,lj_crit,lj_flag
!REAL :: drdp,drdi,dtdp,dtdi,dvdp,dvdi,lj_st,lj_visc
!REAL :: lj_props(10)
lj_tlj_av=0.0_sdk
lj_tvb_av=0.0_sdk
lj_area_av=0.0_sdk
lj_vb_area=0.0_sdk

DO i=1,lj_nx
  DO j=1,lj_nylj
    k=j+(i-1)*lj_ny
    lj_tlj_av(i)=lj_tlj_av(i)+lj_pi*(j+j-1)*lj_dy(k)*lj_dy(k)*lj_tlj(k)
    lj_area_av(i)=lj_area_av(i)+lj_pi*(j+j-1)*lj_dy(k)*lj_dy(k)
  ENDDO
  DO j=lj_nylj+1,lj_ny
    k=j-(i-1)*lj_ny
    l=j-lj_nylj
    lj_tvb_av(i)=lj_tvb_av(i)+lj_pi*(2.0_sdk*lj_rlj(i)*lj_dy(k)+
            & lj_dy(k)*lj_dy(k)*(l*l-1))*lj_tlj(k)
    lj_vb_area(i)=lj_vb_area(i)+lj_pi*(2.0_sdk*lj_rlj(i)*lj_dy(k)+
            & lj_dy(k)*lj_dy(k)*(l*l-1))
  ENDDO
ENDDO

lj_tlj_av=lj_tlj_av/lj_area_av
lj_tvb_av=lj_tvb_av/lj_vb_area

j=lj_nylj
DO i=1,lj_nx
  k=j+(i-1)*lj_ny
  lj_tsurf(i)=1.5_sdk*lj_tlj(k)-0.5_sdk*lj_tlj(k-1)
ENDDO

IF(ttime==0.0)THEN
  rho=lj_rholj(1)
  visc=lj_visc(1)
  lj_re_jet=rho*lj_uin/lj_din/visc
ENDIF

! Theofanous model for jet heat transfer coefficient
IF(lj_re_jet>lj_re_tran)THEN
  ! Turbulent
  i=1
  lj_hhil(i)=0.02*lj_cp(lj_nylj+(i-1)*lj_ny)*
            & lj_rholj(lj_nylj+1+i*(lj_ny+2))*
            & lj_uin*(0.5*i*lj_dx/lj_rlj(i))**(-0.5)
  lj_hil(i)=lj_hhil(i)
  DO i=2,lj_nx
    lj_hhil(i)=0.02*lj_cp(lj_nylj+(i-1)*lj_ny)*
            & lj_rholj(lj_nylj+1+i*(lj_ny+2))*lj_uin*
            & (0.5*i*lj_dx/lj_rlj(i))**(-0.5)
  ENDDO
  DO i=2,lj_nx
    lj_hil(i)=(i*lj_dx*lj_hhil(i)-(i-1)*lj_dx*lj_hhil(i-1))/lj_dx
  ENDDO
ENDIF
lj_hil_av=0.02*lj_cp(lj_nylj)*lj_rholj(lj_nylj+1+lj_ny+2)*
lj_uin*(0.5*lj_nx*lj_dx/lj_rlj(l))**(-0.5)

ELSEIF (lj_re_jet<=lj_re_crit) THEN
  ! Laminar
  j=lj_nylj
  i=1
  k=j+(i-1)*lj_ny
  lj_hhil(i)=4.364*lj_kf(k)/lj_din*(1.0+5.55*exp(-0.13*0.5*i*lj_dx/lj_rlj(l)))
  lj_hil(i)=lj_hhil(i)
  DO i=2,lj_nx
     k=j+(i-1)*lj_ny
     lj_hhil=4.364*lj_kf(k)/lj_din*(1.0+5.55*exp(-0.13*0.5*i*lj_dx/lj_rlj(l)))
  ENDDO
  DO i=2,lj_nx
     lj_hil(i)=(i*lj_dx*lj_hhil(i)-(i-1)*lj_dx*lj_hhil(i-1))/lj_dx
  ENDDO
  lj_hil_av=4.364*lj_kf(lj_nylj)/lj_din !*(0.5*lj_nx*lj_dx/lj_rlj(l))**(-0.5)
ELSE
  ! Transition
  j=lj_nylj
  i=1
  k=j+(i-1)*lj_ny
  ! Laminar component of transition heat transfer coefficient
  lj_hhil(i)=4.364*lj_kf(k)/lj_din*(1.0+5.55*exp(-0.13*0.5*i*lj_dx/lj_rlj(l)))
  lj_hil_lam(i)=lj_hhil(i)
  DO i=2,lj_nx
     k=j+(i-1)*lj_ny
     lj_hhil=4.364*lj_kf(k)/lj_din*(1.0+5.55*exp(-0.13*0.5*i*lj_dx/lj_rlj(l)))
  ENDDO
  DO i=2,lj_nx
     lj_hil_lam(i)=(i*lj_dx*lj_hhil(i)-(i-1)*lj_dx*lj_hhil(i-1))/lj_dx
  ENDDO
  i=1
  k=j+(i-1)*lj_ny
  q=j+1+i*(lj_ny+2)
  ! Turbulent component of transition heat transfer coefficient
  ulam=4000.0*visc/(rho*lj_din)
  lj_hhil(i)=0.02*lj_cp(k)*lj_rholj(q)*ulam*(0.5*i*lj_dx/lj_rlj(l(i)))**(-0.5)
  lj_hil_turb(i)=lj_hhil(i)
  DO i=2,lj_nx
     k=j+(i-1)*lj_ny
     q=j+1+i*(lj_ny+2)
     lj_hhil(i)=0.02*lj_cp(k)*lj_rholj(q)*ulam*(0.5*i*lj_dx/lj_rlj(l(i)))**(-0.5)
  ENDDO
  DO i=2,lj_nx
     lj_hil_turb(i)=(i*lj_dx*lj_hhil(i)-(i-1)*lj_dx*lj_hhil(i-1))/lj_dx
  ENDDO
  lj_hil=lj_hil_lam+(lj_hil_turb-lj_hil_lam)/(lj_re_tran-lj_re_crit)*

}
ENDIF

lj_dtmt_max=0.0

DO i=1,lj_nx
   lj_dtmt_max=lj_dtmt_max+lj_hhil(i)
ENDDO

DO i=1,lj_nx
   k=lj_nylj+(i-1)*lj_ny
   IF(lj_tlj(k).lt.lj_tsat(k))THEN
      lj_hil(i)=lj_hil(i)*DABS((lj_tlj_av(i)-lj_tsat(k))/(lj_tlj(k)-lj_tsat(k)))
   ENDIF
ENDDO

! Young and Bajorek model for noncondensible gas effects
!##############################################################################
!##############################################################################
lj_xcond = 1.0_sdk
IF(lj_gasin > 0.0)THEN
   j=lj_nylj+1
   DO i=1,lj_nx
      k = j+(i-1)*lj_ny
      q = j + 1 + i * (lj_ny+2)
      lj_cfact = 1.2_sdk
      lj_cfact = (lj_pstm_old(lj_ny+2+i*(lj_ny+2))-lj_pstm_old(q))*lj_tsat(k) / &
      (lj_pres(k)*lj_tlj(lj_ny+i*(lj_ny))-lj_tlj(k))
      lj_renew_time = 0.5_sdk*lj_rlj(i)*0.5_sdk*(lj_ulj(k-1)+lj_ulj(k-1+lj_ny))
      lj_xcond(i) = 0.5_sdk*((DABS(lj_afact * lj_afact - 4.0_sdk * &
         lj_afact))**0.5 - lj_afact)
      IF(lj_ulj(k) == 0.0) lj_ulj(k) = lj_ulj(k-1)*1.0
      lj_xcond(i) = 0.168*((lj_gasin*lj_rholj(q-1)*lj_ulj(k-1)/ &
         (lj_rholj(q)*lj_ulj(k)))**(-0.1)) &
         *(0.5*i*lj_dx/lj_rlj(i))**(0.33)
   ENDDO
ENDIF

!##############################################################################
!##############################################################################
lj_hil=lj_hil*lj_xcond

! Model from Young and Bajorek for mass transfer coefficient
lj_gij=0.0_sdk
lj_qil=0.0_sdk
DO i=1,lj_nx
  k=j+(i-1)*lj_ny
  q=(j+1)+i*(lj_ny+2)
  lj_qil(k)=2.0*lj_hil(i)*lj_pi*2.0_sdk*lj_rlj(i)*lj_dx/lj_cp(k)
  lj_qil(k)=1.0*lj_hil(i)*lj_pi*2.0*lj_rlj(i)*lj_dx/lj_cp(k)
  IF(lj_ilj_old(q).gt.lj_if(k)) THEN
    lj_qil(k)=0.0_sdk
  ENDIF
  lj_glj(q)=-lj_qil(k)/(lj_ilj_old(q+1)-lj_if(k+1))
  lj_glj(q+1)=lj_glj(q)
ENDDO

DO i=1,lj_nx
  k=(lj_nylj+1)+(i-1)*lj_ny
  q=lj_nylj+2+i*(lj_ny+2)
  lj_vi(i)=0.5*lj_glj(q)/(lj_pi*lj_rlj(i)*lj_dx*lj_rholj(q))
  WRITE(17,*) lj_vi(i)
ENDDO

lj_vif=0.0_sdk
!
! Calculate the max time step base on mass transfer
!
ENDDO

lj_dtmt_max=MINVAL(lj_dtmt)
!
lj_dhildp=lj_drhodp
lj_dqildp=lj_drhodp*lj_difdp
!
END SUBROUTINE liqsideh

SUBROUTINE jetshear
IMPLICIT NONE
REAL :: lj_fi,lj_rshear
INTEGER :: i,j,k,q
lj_fi=0.005_sdk*(1+75)
lj_kidrag=0.0_sdk

j=lj_nylj+1
DO i=1,lj_nx+1
  IF(i.ne.lj_nx+1) THEN
    lj_rshear=lj_rlj(i)
  ELSE
    lj_rshear=0.0_sdk
  ENDIF
  lj_dtmt(i)=DABS(lj_rholj(q)*lj_vol(k)/(lj_qil(k)-lj_glj(q)*lj_if(k+1)))
ENDDO

ELSE
  lj_rshear=lj_rlj(i-1)
ENDIF

k=j+(i-1)*lj_ny
q=j+1+i*(lj_ny+2)

lj_kidrag(k)=-4.0_sdk*lj_fi/lj_rshear*lj_rholj(q)*
         & DABS(lj_ulj_old(q-1)-lj_ulj_old(q))
ENDDO

! lj_kidrag=0.0_sdk

END SUBROUTINE jetshear

END MODULE JetHeatTrans
MODULE LiqJetSolve

! Iterative solvers for 2-D Conduction Problem

USE JetIntrType
USE JetDataIn
USE JetLUsolve
USE JetSetEqn
USE JetSetVol

CONTAINS

SUBROUTINE Implic

IMPLICIT NONE

INTEGER(sik) :: info,k,i,j,q
REAL(sdk),ALLOCATABLE :: ai(:,:),bi(:),bd(:)

IF (band) THEN
  ALLOCATE (ai(lj_ldi,lj_k),bi(lj_k),bd(lj_k))
ELSE
  ALLOCATE (ai(lj_k,lj_k),bi(lj_k),bd(lj_k))
ENDIF

CALL JetAxMom

! OPEN(19,FILE='axmom.csv')
! DO i=1,ljNx+1
!   DO j=1,lj_ny
!     k=j+(i-1)*lj_ny
!     WRITE(19,*) i,',',j,',',lj_u0(k),',',lj_c0(k)
!   ENDDO
! ENDDO

CALL JetRadMom

! OPEN(21,FILE='radmom.csv')
! DO i=1,lj_nx
!   DO j=1,lj_ny+1
!     k=j+(i-1)*(lj_ny+1)
!     WRITE(21,*) i,',',j,',',lj_v0(k),',',lj_d0(k)
!   ENDDO
! ENDDO

CALL JetNonCon
CALL TwoJetLHS

! OPEN(13,FILE='matrix.csv')
! OPEN(26,FILE='matrix1.csv')
! OPEN(39,FILE='matrix2.csv')
! DO i=1,lj_k
!  DO j=1,lj_k
!    IF (lj_z(i,j).ne.0.0_sdk) THEN
!      k=CEILING(0.333333333333333333333333333333333333333_sdk*i)
!      IF (MOD(i,3).eq.1) THEN
!        WRITE(39,'(I4,a,I4,a,E12.6,a,E12.6,a,E12.6)') i,',',j,',',lj_z(i,j),',',lj_r(k),',',lj_z(i,j)/lj_r(k)
!      ELSEIF (MOD(i,3).eq.2) THEN
!        WRITE(13,'(I4,a,I4,a,E12.6,a,E12.6,a,E12.6)') i,',',j,',',lj_z(i,j),',',lj_r(k),',',lj_z(i,j)/lj_r(k)
!      ELSE
!        WRITE(26,'(I4,a,I4,a,E12.6,a,E12.6,a,E12.6)') i,',',j,',',lj_z(i,j),',',lj_r(k),',',lj_z(i,j)/lj_r(k)
!      ENDIF
!    ENDIF
!  ENDDO
! ENDDO

CALL TwoJetRHS

! OPEN(14,FILE='b.csv')
! DO i=1,lj_k
!  k=CEILING(0.333333333333333333333333333333333333333_sdk*i)
!  WRITE(14,'(I4,a,E12.6,a,E12.6,a,E12.6)') i,',',lj_b(i),',',lj_r(k),',',lj_b(i)/lj_r(k)
! ENDDO

ai=lj_z
bi=lj_b

! bde=-1.0_sdk*(lj_x_old-bi)
bd=bi

IF (band) THEN
  CALL dgbfa(ai,lj_ldi,lj_k,lj_wl,lj_wu,lj_ipvt,info)
  CALL dgbsl(ai,lj_ldi,lj_k,lj_wl,lj_wu,lj_ipvt,bd,0)
ELSE
  CALL dgefa(ai,lj_k,lj_k,lj_ipvt,info)
  CALL dgesl(ai,lj_k,lj_k,lj_ipvt,bd,0)
ENDIF

lj_x=bd

DO k=1,lj_n
  lj_pres(k)=lj_x(k+k+k-1)
  lj_ilj(k)=lj_x(k+k+k)
  lj_gas(k)=lj_x(k+k+k-2)
ENDDO

lj_pgas = lj_gas * lj_pres
lj_pstm = lj_pres - lj_pgas

IF (ttime.eq.0.0) THEN
  lj_en_old=0.0_sdk
  DO i=1,lj_nx
    DO j=1,lj_nylj
      k=j+(i-1)*lj_ny
      lj_x(k)=lj_x(k)+lj_pgas
    ENDDO
  ENDDO
ENDIF
\[ q = j + 1 + i \cdot \left( l_{j_ny} + 2 \right) \]

\[ l_{j_en \_old} = l_{j\_en \_old} + l_{j\_rhollj}\left(q\right) \cdot l_{j\_vol}\left(k\right) \cdot l_{j\_ilj\_old}\left(q\right) \]

ENDDO

ENDDO

ENDIF

DO \ i = 1, l_{j_nx} 
  DO \ j = 1, l_{j_ny} 
    k = j + \left( i - 1 \right) \cdot l_{j_ny} 
    q = j + 1 + i \cdot \left( l_{j_ny} + 2 \right) 

    \begin{align*} 
    l_{j\_err}\left(k\right) &= \left( l_{j\_ilj}\left(k\right) - l_{j\_ilj\_old}\left(q\right) \right) / l_{j\_ilj\_old}\left(q\right) \cdot 100.0 \_sdk \\
    l_{j\_edif}\left(k\right) &= l_{j\_ilj}\left(k\right) - l_{j\_ilj\_old}\left(q\right) \\
    l_{j\_perr}\left(k\right) &= \left( l_{j\_pres}\left(k\right) - l_{j\_pres\_old}\left(q\right) \right) / l_{j\_pres\_old}\left(q\right) \cdot 100.0 \_sdk \\
    l_{j\_pdif}\left(k\right) &= l_{j\_pres}\left(k\right) - l_{j\_pres\_old}\left(q\right) 
    \end{align*} 

  \end{align*} 

ENDDO

ENDDO

DO \ i = 1, l_{j_nx} 
  DO \ j = 1, l_{j_ny} 
    k = j + \left( i - 1 \right) \cdot l_{j_ny} 
    q = \left( j + 1 \right) + i \cdot \left( l_{j_ny} + 2 \right) 

    \begin{align*} 
    l_{j\_pres\_old}\left(q\right) &= l_{j\_pres}\left(k\right) \\
    l_{j\_ilj\_old}\left(q\right) &= l_{j\_ilj}\left(k\right) \\
    l_{j\_pstm\_old}\left(q\right) &= l_{j\_pstm}\left(k\right) \\
    l_{j\_pgas\_old}\left(q\right) &= l_{j\_pgas}\left(k\right) \\
    l_{j\_gas\_old}\left(q\right) &= l_{j\_gas}\left(k\right) 
    \end{align*} 

ENDDO

ENDDO

! Set Dummy value at \ j = 0
DO \ i = 1, l_{j_nx} 
  q = 1 + i \cdot \left( l_{j_ny} + 2 \right) 
  k = 2 + i \cdot \left( l_{j_ny} + 2 \right) 

  \begin{align*} 
  l_{j\_pres\_old}\left(q\right) &= l_{j\_pres\_old}\left(k\right) \\
  l_{j\_ilj\_old}\left(q\right) &= l_{j\_ilj\_old}\left(k\right) \\
  l_{j\_pstm\_old}\left(q\right) &= l_{j\_pstm\_old}\left(k\right) \\
  l_{j\_pgas\_old}\left(q\right) &= l_{j\_pgas\_old}\left(k\right) \\
  l_{j\_gas\_old}\left(q\right) &= l_{j\_gas\_old}\left(k\right) 
  \end{align*} 

ENDDO

! Set Dummy value at \ j = l_{j_ny} + 1
DO \ i = 1, l_{j_nx} 
  q = l_{j_ny} + 2 + i \cdot \left( l_{j_ny} + 2 \right) 
  k = l_{j_ny} + 1 + i \cdot \left( l_{j_ny} + 2 \right) 
  j = l_{j_ny} + i \cdot \left( l_{j_ny} + 2 \right) 

  ! \begin{align*} 
  l_{j\_pres\_old}\left(q\right) &= 2.0 \_sdk \cdot l_{j\_pres\_old}\left(k\right) - l_{j\_pres\_old}\left(j\right) \\
  l_{j\_ilj\_old}\left(q\right) &= 2.0 \_sdk \cdot l_{j\_ilj\_old}\left(k\right) - l_{j\_ilj\_old}\left(j\right) 
  \end{align*} 

ENDDO

! Set Dummy value at \ i = 0
DO \ j = 1, l_{j_nylj} 
  q = \left( j + 1 \right) + 0 \cdot \left( l_{j_ny} + 2 \right) 
  k = \left( j + 1 \right) + 1 \cdot \left( l_{j_ny} + 2 \right) 
  i = \left( j + 1 \right) + 2 \cdot \left( l_{j_ny} + 2 \right) 

  ! \begin{align*} 
  l_{j\_pres\_old}\left(q\right) &= 2.0 \_sdk \cdot l_{j\_pres\_old}\left(k\right) - l_{j\_pres\_old}\left(i\right) 
  \end{align*}
lj_ilj_old(q) = 2.0 * sdk * lj_ilj_old(k) - lj_ilj_old(i)
ENDDO

DO j = lj_nylj + 1, lj_ny
    q = (j + 1) + 0 * (lj_ny + 2)
    k = (j + 1) + 1 * (lj_ny + 2)
    i = (j + 1) + 2 * (lj_ny + 2)
    lj_ilj_old(q) = 2.0 * sdk * lj_ilj_old(k) - lj_ilj_old(i)
    lj_pres_old(q) = 2.0 * sdk * lj_pres_old(k) - lj_pres_old(i)
ENDDO

Set Dummy value at i = lj_nx + 1
DO j = 1, lj_ny
    q = (j + 1) + (lj_nx + 1) * (lj_ny + 2)
    k = (j + 1) + (lj_nx) * (lj_ny + 2)
    i = (j + 1) + (lj_nx - 1) * (lj_ny + 2)
    lj_ilj_old(q) = 2.0 * sdk * lj_ilj_old(k) - lj_ilj_old(i)
    lj_pres_old(q) = 2.0 * sdk * lj_pres_old(k) - lj_pres_old(i)
ENDDO
CALL JetProps

DO i = 1, lj_nx + 1
    DO j = 1, lj_ny
        k = j + (i - 1) * lj_ny
        q = (j + 1) + i * (lj_ny + 2)
        IF (j .eq. lj_nylj + 1) THEN
            lj_ulj(k) = (1.0 * sdk + lj_kidrag(k) * lj_dt) * (lj_u0(k) +
                        lj_c0(k) * (lj_pres_old(q) - lj_pres_old(q - (lj_ny + 2))) +
                        lj_cl(k) * (lj_pres_old(q) + lj_pres_old(q - (lj_ny + 2))) +
                        lj_c2(k) * (lj_ilj_old(q) + lj_ilj_old(q - (lj_ny + 2)))) -
                        lj_ulj_old(q - 1) * lj_kidrag(k) * lj_dt
        ELSE
            lj_ulj(k) = lj_u0(k) + lj_c0(k) * (lj_pres_old(q) -
                        lj_pres_old(q - (lj_ny + 2))) +
                        lj_cl(k) * (lj_pres_old(q) + lj_pres_old(q - (lj_ny + 2))) +
                        lj_c2(k) * (lj_ilj_old(q) + lj_ilj_old(q - (lj_ny + 2)))
        ENDIF
    ENDDO
ENDDO

DO i = 1, lj_nx
    DO j = 1, lj_ny + 1
        k = j + (i - 1) * (lj_ny + 1)
        q = (j + 1) + i * (lj_ny + 3)
        lj_vlj(k) = lj_v0(k) + lj_d0(k) * (lj_pres_old(q - i) - lj_pres_old(q - 1 - i)) +
                    lj_d1(k) * (lj_pres_old(q - i) + lj_pres_old(q - 1 - i)) +
                    lj_d2(k) * (lj_ilj_old(q - i) + lj_ilj_old(q - 1 - i))
    ENDDO
ENDDO

DO i = 1, lj_nx + 1
    DO j = 1, lj_ny
        k = j + (i - 1) * lj_ny
        q = j + i * (lj_ny + 2)
        IF (lj_ulj_old(q) .ne. 0.0) THEN
            lj_uerr(k) = (lj_ulj(k) - lj_ulj_old(q)) / lj_ulj_old(q) * 100.0 * sdk
ENDIF
ENDDO
ENDDO
DO i=1,lj_nx
DO j=1,lj_ny+1
  k=j+(i-1)*lj_ny
  q=j+1+i*(lj_ny+3)
  IF(lj_vlj_old(q).ne.0.0)THEN
    lj_verr(k)=(lj_vlj(k)-lj_vlj_old(q))/lj_vlj_old(q)*100.0_sdk
  ENDIF
ENDDO
ENDDO
DO i=1,lj_nx+1
  DO j=1,lj_ny
    k=j+(i-1)*lj_ny
    q=j+i*(lj_ny+2)
    lj_ulj_old(q)=lj_ulj(k)
  ENDDO
ENDO
DO i=1,lj_nx
  DO j=1,lj_ny+1
    k=j+(i-1)*lj_ny
    q=j+1+i*(lj_ny+3)
    lj_vlj_old(q)=lj_vlj(k)
  ENDDO
ENDO
! Set Dummy cells for axial momentum
DO i=1,lj_nx+1
  q=1+i*(lj_ny+2)
  k=2+i*(lj_ny+2)
  lj_ulj_old(q)=lj_ulj_old(k)
ENDDO
DO i=1,lj_nx+1
  q=1+lj_ny+2+i*(lj_ny+2)
  k=1+lj_ny+1+i*(lj_ny+2)
  j=1+lj_ny+i*(lj_ny+2)
  lj_ulj_old(q)=2.0_sdk*lj_ulj_old(k)-lj_ulj_old(j)
ENDDO
DO j=1,lj_ny+2
  q=j+0*(lj_ny+2)
  k=j+1*(lj_ny+2)
  i=j+2*(lj_ny+2)
  lj_ulj_old(q)=2.0_sdk*lj_ulj_old(k)-lj_ulj_old(i)
ENDDO
DO j=1,lj_ny+2
  q=j+(lj_nx+2)*(lj_ny+2)
  k=j+(lj_nx+1)*(lj_ny+2)
  i=j+lj_nx*(lj_ny+2)
lj_ulj_old(q)=2.0_sdk*lj_ulj_old(k)-lj_ulj_old(i)
ENDDO

! Set Dummy cells for radial momentum
DO i=1,lj_nx
  q=1+i*(lj_ny+3)
  k=3+i*(lj_ny+3)
  lj_vlj_old(q)=-1.0_sdk*lj_vlj_old(k)
ENDDO

DO i=1,lj_nx
  q=lj_ny+2+i*(lj_ny+3)
  k=lj_ny+1+i*(lj_ny+3)
  j=lj_ny+i*(lj_ny+3)
  lj_vlj_old(q)=2.0_sdk*lj_vlj_old(k)-lj_vlj_old(j)
ENDDO

DO j=1,lj_ny+1
  q=j+0*(lj_ny+3)
  k=j+1*(lj_ny+3)
  i=j+2*(lj_ny+3)
  lj_vlj_old(q)=2.0_sdk*lj_vlj_old(k)-lj_vlj_old(i)
ENDDO

DO j=1,lj_ny+1
  q=j+(lj_nx+1)*(lj_ny+3)
  k=j+(lj_nx)*(lj_ny+3)
  i=j+(lj_nx-1)*(lj_ny+3)
  lj_vlj_old(q)=2.0_sdk*lj_vlj_old(k)-lj_vlj_old(i)
ENDDO
END SUBROUTINE Implic

END MODULE LiqJetSolve
B.2 Coupling Subroutine (liquid_jet_int.f90)

MODULE liquid_jet_int

USE JetDataIn

REAL :: djet,ljet,tjet,ujet,dvb,tvb, mass_out, energy_out, tedit, editdt
REAL, ALLOCATABLE :: lj_cobra_gamma(:), lj_gamma(:), cobra_map(:,), jet_mmult(:,)
REAL, ALLOCATABLE :: nodesize_map(:)
INTEGER, ALLOCATABLE :: jet_map(:,)
INTEGER :: nx, nylj, nyvb, mjet, jjet, no_nodes, maxnodes
INTEGER :: start_node, end_node, first_node, last_node

CONTAINS

SUBROUTINE init_liquid_jet(icobra,djet,ljet,tjet,ujet,dvb,tvb,nx,nylj,nyvb)

use spltdat, only: lchan, dxs
use public_dimensions, only: mxpub
use JetOutput, only: Echo

IMPLICIT NONE

INTEGER :: jet_section, i, j, k, n, q, r, s
INTEGER, INTENT(IN) :: icobra, nx, nylj, nyvb
REAL, INTENT(IN) :: djet, ljet, tjet, ujet, dvb, tvb
REAL :: nodesize, length, r_frac, end_frac, num_jet_nodes

! Initialize the variable arrays in JetDataIn
lj_din = djet
lj_length = ljet
lj_tin = tjet
lj_uin = ujet
lj_dvb = dvb
lj_tstm = tvb
lj_nx = nx
lj_nylj = nylj
lj_nyvb = nyvb
lj_pin = 0.0_sdk
lj_gas_in = 0.0_sdk
lj_tstart = 0.0_sdk
ttime = lj_tstart
tedit = 0.0_sdk
editdt = 0.2_sdk
tedit = tedit + editdt

CALL SetData

open(11, FILE='jetout.csv')
close(11)

CALL Echo
! Need to link here how cobra cells are linked to jet noding

jet_section=lchan(mjet) ! get section number for jet location

length = 0.0

j = jjet+1
n = 0

do while(length < lj_length)
  j = j - 1
  IF(j == 0) then
    WRITE(*,*) "jet length exceeds allowable in channel"
    STOP
  ENDIF

  nodesize = dxs(jet_section, j)
  length = length + nodesize

  n = n + 1
endo

no_nodes = n

end_frac = 0.0
if(length > lj_length) then
  end_node = j
  end_frac = length - lj_length
endif

if(length == lj_length) end_node = j - 1

last_node = j
first_node = jjet

! Allocate the arrays based on the size of the jet problem and
! the number of nodes needed for the jet
ALLOCATE (lj_gamma(lj_nx), lj_cobra_gamma(mxpub), jet_map(2,lj_nx))
ALLOCATE (jet_mmult(2,lj_nx), nodesize_map(mxpub), cobra_map(mxpub))

jet_mmult = 0.0
jet_map = 0
cobra_map = 0.0

! Map the cobra noding to the jet model noding
length = 0.0
j=jjet + 1
do i=1, no_nodes
  j = j - 1
  nodesize = dxs(jet_section, j)
  length = length + nodesize
  nodesize_map(j) = nodesize ! length of each node
  cobra_map(j) = length ! total length at the end of each cobra node
endo

nodesize_map(jjet+1-no_nodes) = nodesize_map(jjet+1-no_nodes) - end_frac
cobra_map(jjet+1-no_nodes) = cobra_map(jjet+1-no_nodes) - end_frac

q = 0
r = 0
r_frac = 0.0
s = 1
j = jjet + 1
do i=1,no_nodes
  j = j - 1
  num_jet_nodes = (nodesize_map(j)-r_frac*lj_dx) / lj_dx
  q = floor(num_jet_nodes)
  r_frac = 1.0 - (num_jet_nodes - q)
  if(r_frac < 1.0) then
    r = ceiling(num_jet_nodes) + s - 1
    do n = s, (q+s-1)
      jet_map(1,n) = j
      jet_mmult(1,n) = 1.0
    enddo
    jet_map(1,r) = j
    jet_map(2,r) = j - 1
    jet_mmult(1,r) = 1.0 - r_frac
    jet_mmult(2,r) = r_frac
    s = r + 1
  else
    do n = s, (num_jet_nodes+s-1)
      jet_map(1,n) = j
      jet_mmult(1,n) = 1.0
    enddo
    s = num_jet_nodes + s - 1 + 1
  endif
enddo
!do i=1,lj_nx
!  write(*,*) "jet map",i,jet_map(1,i)
!enddo

END SUBROUTINE init_liquid_jet

SUBROUTINE move_jet_old

IMPLICIT NONE

lj_pres_old=lj_p_rst
lj_ilj_old=lj_i_rst
lj_pgas_old=lj_ppa_rst
lj_pstm_old=lj_pps_rst
lj_tdp_old=lj_tdp_rst
lj_ulj_old=lj_u_rst
lj_vlj_old=lj_v_rst
lj_gas_old=lj_gas_rst

END SUBROUTINE move_jet_old

SUBROUTINE move_jet_new

IMPLICIT NONE

lj_p_rst=lj_pres_old
lj_i_rst=lj_ilj_old
lj_ppa_rst=lj_pgas_old
lj_pps_rst=lj_pstm_old
lj_tdp_rst=lj_tdp_old
lj_u_rst=lj_ulj_old
lj_v_rst=lj_vlj_old

END SUBROUTINE move_jet_new
lj_gas_rst= lj_gas_old

END SUBROUTINE move_jet_new

SUBROUTINE jet_heat_transfer

use LiqJetSolve, only: Implic
use contrlr, only: delt
use mcmx, only : p
use gasvar, only : pmgas, pst
use twophas, only : hv
use JetOutput, only: Edit,PrintOut
use JetSetVol, only: JetCellSize,JetProps,JetDiffusivity,JetOneSet
use JetHeatTrans, only: liqsideh, jetshear

IMPLICIT NONE

real :: cdt,umax
integer :: i,j,k,q

! get cobra time step size
cdt = delt
write(*,*) "cobra time step", cdt
lj_tmax = 0.5*cdt
! tedit = 0.0
! lj_tstart = 0.0
lj_tstart = ttime
lj_tend = lj_tstart + cdt

do i=1,lj_nx
  j=1lj_ny+1
  q = j + 1 + i * (lj_ny + 2)
  write(*,*) "jet pressure", lj_pres_old(q), "cobra pressure", p(mjet,jet_map(1,i))
  write(*,*) "jet enthalpy", lj_ilj_old(q), "cobra enthalpy", hv(mjet,jet_map(1,i))
  lj_pres_old(q) = p(mjet,jet_map(1,i))
  write(*,*) "cobra pressure", lj_pres_old(q)
  lj_ilj_old(q) = hv(mjet,jet_map(1,i))
  write(*,*) "cobra enthalpy", lj_ilj_old(q)
enddo

! set vapor variables to cobra state
! do j=lj_nylj+1,lj_ny+1
!   i=1lj_nx+1
!   q=j+1+i*(lj_ny +2)
!   lj_pres_old(q)=lj_pres_old(lj_ny+2+(i)*(lj_ny+2))
!   i=0
!   q=j+1+i*(lj_ny +2)
!   lj_ilj_old(q)=lj_ilj_old(lj_ny+2+(i)*(lj_ny+2))
! enddo

! ttime = lj_tstart
lj_gamma = 0.0
lj_cobra_gamma = 0.0
mass_out = 0.0
energy_out = 0.0

CALL JetCellSize
CALL liqsideh

DO WHILE (ttime < lj_tend)
  !
  CALL JetCellSize ! Set the size for computational volumes
  CALL JetProps ! Set fluid properties of jet and vapor boundary
  CALL JetDiffusivity
  CALL liqsideh
  CALL jetshear
  ! Insert time step control here******************************************************************************

  umax=DABS(MAXVAL(lj_ulj))
  !
  lj_dt=lj_dx/4800.0_sdk
  lj_dt=0.5_sdk*lj_dx/umax
  lj_vdt=0.0_sdk
  DO i=1,lj_nx
    DO j=1,lj_ny+1
      k=j+(i-1)*(lj_ny+1)
      IF(lj_vlj(k).ne.0.0)THEN
        lj_vdt(k)=DABS(lj_dyy(k)/lj_vlj(k))
      ELSE
        lj_vdt(k)=1000.0_sdk
      ENDIF
    ENDDO
    ENDDO
  lj_difhxdt=1000.0_sdk
  lj_difhrdt=1000.0_sdk
  lj_difmxdt=1000.0_sdk
  lj_difmrdt=1000.0_sdk
  DO i=1,lj_nx
    DO j=1,lj_nylj
      k=j+(i-1)*lj_ny
      lj_difhxdt(k)=0.5*lj_dx*lj_dx/lj_edh(k)
      lj_difhrdt(k)=0.5*lj_dy(k)*lj_dy(k)/lj_edh(k)
      lj_difmxdt(k)=0.5*lj_dx*lj_dx/lj_edm(k)
      lj_difmrdt(k)=0.5*lj_dy(k)*lj_dy(k)/lj_edm(k)
    ENDDO
    ENDDO
  lj_difhxdt_max=MINVAL(lj_difhxdt)
  lj_difhrdt_max=MINVAL(lj_difhrdt)
  lj_difmxdt_max=MINVAL(lj_difmxdt)
  lj_difmrdt_max=MINVAL(lj_difmrdt)
  lj_vdt_max=MINVAL(lj_vdt)
  IF(lj_dt.gt.lj_dtmt_max)THEN
    lj_dt=lj_dtmt_max
  ENDIF
  IF(lj_dt.gt.lj_vdt_max.and.lj_vdt_max.gt.0.0)THEN
    lj_dt=lj_vdt_max
  ENDIF
  IF(lj_dt.gt.lj_difhxdt_max)THEN
    lj_dt=lj_difhxdt_max
  ENDIF
  IF(lj_dt.gt.lj_difhrdt_max)THEN
    lj_dt=lj_difhrdt_max
  ENDIF
  IF(lj_dt.gt.lj_difmxdt_max)THEN
    lj_dt=lj_difmxdt_max
  ENDIF
  IF(lj_dt.gt.lj_difmrdt_max)THEN
    lj_dt=lj_difmrdt_max
  ENDIF
lj_dt = lj_difhrdt_max
ENDIF
IF(lj_dt.gt.lj_difmxdt_max)
  lj_dt = lj_difmxdt_max
ENDIF
IF(lj_dt.gt.lj_difmrdt_max)
  lj_dt = lj_difmrdt_max
ENDIF
IF(lj_dt.gt.lj_tmax) THEN
  lj_dt = lj_tmax
ENDIF
!  lj_dt = 0.1_sdk * lj_dt
DO k = 1, lj_l
  IF((lj_kidrag(k)*lj_dt).lt.-0.75_sdk) THEN
    lj_kidrag(k) = -0.75_sdk/lj_dt
  ENDIF
ENDDO
lj_wev = lj_rholj(lj_nylj+2)*lj_uin*lj_uin*lj_din/lj_st(1)
lj_wef = lj_rholj(1)*lj_uin*lj_uin*lj_din/lj_st(1)
******************************************************************************
CALL Implic
CALL JetProps
CALL JetOneSet
j = lj_nylj
DO i = 1, lj_nx
  q = j + i*(lj_ny+2)
  k = j + (i-1)*lj_ny
  lj_gamma(i) = lj_gamma(i) + lj_glj(q) * (lj_if(k+1)-lj_ilj(k)) * lj_dt
ENDDO
! Calculate the mass and energy into the dump node at the end of jet
i = lj_nx + 1
DO j = 1, lj_nylj
  k = j + (i-1) * lj_ny
  q = j + 1 + i * (lj_ny+2)
  mass_out = mass_out + lj_ulj(k) * lj_axlj(k) * lj_rholj(q-(lj_ny+2)) * lj_dt
  energy_out = energy_out + lj_ulj(k) * lj_axlj(k) * lj_rholj(q-(lj_ny+2)) * lj_ilj(k-lj_ny) * lj_dt
ENDDO

if(ttime.ge.tedit) then
  call Edit
  call PrintOut
  tedit = tedit + editdt
endif
ENDDO  ! End of do while loop
!
! Calculate the mass out of the cobra cells due to mass transfer
DO i = 1, lj_nx
  j = jet_map(1,i)
  k = jet_map(2,i)

  lj_cobra_gamma(j) = lj_cobra_gamma(j) + lj_gamma(i) * jet_mmult(1,i)
  IF(k > 0) then
    lj_cobra_gamma(k) = lj_cobra_gamma(k) + lj_gamma(i) * jet_mmult(2,i)
  endif
ENDDO

! Divide mass and energy terms by cobra time step size
mass_out = mass_out / cdt
energy_out = energy_out / cdt
lj_cobra_gamma = lj_cobra_gamma / cdt

END SUBROUTINE jet_heat_transfer

SUBROUTINE add_jet_mass(i,j)

  use simsol, only: airs
  use twophas, only: hv
  use JetDataIn, only: lj_uin,lj_din

  IMPLICIT NONE

  INTEGER, INTENT(IN) :: i,j  ! channel and node number

  if (i == mjet) then
    if (last_node <= j .and. j <= first_node) then

      ! vapor mass removed from cobra cell
      airs(6) = airs(6) + lj_cobra_gamma(j)

      ! vapor energy removed from cobra cell
      airs(3) = airs(3) + lj_cobra_gamma(j) * hv(mjet,j)
      write(*,*) "mass added", lj_cobra_gamma(j)
      write(*,*) "energy added", lj_cobra_gamma(j) * hv(mjet,j)
    endif
    if (j == end_node) then
      ! airs(6) = ! vapor boundary layer mass at jet end
      ! airs(3) = ! vapor boundary layer energy at jet end
      airs(2) = airs(2) - mass_out  ! jet mass at jet end
      airs(4) = airs(4) - energy_out  ! jet energy at jet end
      write(*,*) "liquid mass", mass_out, lj_uin*lj_din*lj_din*3.14/4.0*62.41
    endif
  endif

END SUBROUTINE add_jet_mass

END MODULE liquid_jet_int
Francis Xavier Buschman III was born on March 23, 1980, in Lansdale, Pennsylvania. He attended Pennridge High School in Perkasie, Pennsylvania, from which he graduated in 1998. He then attended the Pennsylvania State University and received his Bachelor of Science degree in Nuclear Engineering in 2002. While attending college he worked as a Reactor operator for the Breazeale Nuclear Reactor at the Radiation Science and Engineering Center. Following graduation, he enrolled as a graduate student at the Pennsylvania State University studying Nuclear Engineering. While in graduate school he studied under the Naval Nuclear Propulsion Fellowship Program. His research while in graduate school has focused on the areas of jet flow, condensation heat transfer, and the development of a liquid jet model for predicting condensation on water jets.